NR INDEKSU 350965
PL ISSN 0860-1674

12

Z MIKROKOMPUTEREM NA TY

Bajtek

MIESIĘCZNY DODATEK DO SZTANDARU MŁODYCH NR 12 GRUDZIEŃ 1986 CENA 100 ZŁ



KIERUNEK: KSIĘŻYC!



PROGRAMIŚCI JUTRA

Czy widzieliście kiedyś bajkowe miasteczko? Ja widziałem. Zbudowane było ze śniegu i lodu, udekorowane świerkowymi choinkami, obwieszone kolorowymi lampionami. Czego tam nie było — lodowe wieże i zamki obronne kilkunastometrowej wysokości, śnieżne labirynty, potężny Dziadek Mróz i Śnieżynka, stumetrowej długości ślizgawka, dużo muzyki, radości i niemilknącego długo w nocy gwaru. Raj dla wszystkich dzieci. To bajkowe miasteczko (czyli „skazocznyj gorodok") zbudowane zostało na najruchliwszej ulicy Akadiemgorodka — stolicy syberyjskich uczonych. Ruch uliczny został tam oczywiście na czas zimy zamknięty. To nie jest chyba źle, że radość dzieci wzięła górę nad wygodą kierowców, zmuszonych wykonywać kręte objazdy. Narzekań jednak nie słyszałem — może dlatego, że kierowcy też kiedyś byli dziećmi, a bajkowe miasteczka buduje się na Syberii co roku.

W pierwszym tegorocznym numerze „Bajtka", kończąc reportaż o doświadczeniach profesora Andrieja Jerszowa w nauczaniu informatyki uczniów nowosybirskich szkół, napisałem, że do Akadiemgorodka jeszcze na tych łamach wrócimy. Właśnie to czynię. Jest zresztą ku temu świetna okazja. Oto bowiem „Komsomolska Prawda" — gazeta radzieckiej młodzieży — w dniu 29 października 1986 roku zamieściła na pierwszej stronie informację pod tytułem „Programiści przyszłości", z której można się dowiedzieć, że:

„Wczoraj, na zaproszenie amerykańskiej fundacji „Nietknięta ziemia" grupa nowosybirskich uczniów odleciała samolotem Aerofłotu do USA. Nie jest to grupa zwykła. Pod kierunkiem akademika A. Jerszowa dzieci te od kilku lat zajmują się informatyką, uczą się współpracy z komputerem. W programie wyjazdu — poznanie głównych komputerowych centrów Ameryki, wspólne zajęcia z amerykańskimi rówieśnikami w szkolnych laboratoriach informatyki... Jest to pierwszy służbowy wyjazd radzieckich uczniów do USA".

Oczywiście, w tym przypadku nie naukowy program wyjazdu jest najważniejszy (choć to dobra wiadomość), że dzięki informatyce można zwiedzać świat! Andriej Jerszow wyjaśnia: „Współczesne elektroniczne maszyny cyfrowe są nie tylko instrumentem służącym rozwiązywaniu zadań ściśle naukowych, ale i uniwersalnym środkiem międzyludzkiego współżycia. Nasze i amerykańskie dzieci powinny spotykać się ze sobą, powinny już dzisiaj współpracować w imię przyszłości. Dlatego że jutro wspólnie mają przekroczyć granicę trzeciego tysiąclecia, wspólnie przyjmować na siebie ciężar odpowiedzialnych politycznych decyzji. Jedziemy do Ameryki, aby te szczere, przyjacielskie więzi, które mam nadzieję, zawiążą się w czasie spotkań u naszych i amerykańskich dzieci, przenieść bez strat przez najbliższe piętnastolecie i z tymi uczuciami wzajemnego zaufania wkroczyć w XXI wiek".

Podobnej wypowiedzi udzielił dziennikarzowi „Komsomolskiej Prawdy" organizator wyjazdu ze strony amerykańskiej, dyrektor fundacji „Nietknięta ziemia" D. Perri: „Mamy nadzieję, że ta pierwsza służbowa wizyta grupy syberyjskich uczniów w Ameryce i czekający nas w ramach rewizyty wyjazd amerykańskich dzieci do ZSRR odegrają swoją rolę w szlachetnym dziele uzyskiwania wzajemnego zaufania".

Czy rzeczywiście dzieci mogą realnie przyczynić się do zachowania pokoju? Znany amerykański futurolog Alvin Toffler twierdzi, że robienie polityki już dawno przestało być li tylko sprawą polityków. Każdy z nas, tworząc to, co nazywamy opinią publiczną, wywiera mniejszy lub większy wpływ na bieg wydarzeń. Czasami jest to wpływ bardzo duży. Nawet podróże wielu polityków nie uzyskały takiego rozgłosu na świecie jak przyjazd małej Samanthy Smith do Związku Radzieckiego a potem rewizyta Katii Łyczewej w USA. Samantha tragicznie zginęła w katastrofie lotniczej, ale wytyczona przez nią drogą odbyło już podróże przyjaźni kilkuset małych Amerykanów i Rosjan. Młodzi informatycy z Nowosybirska są w tej sztafecie kolejną zmianą. A może by tak, zapytam na marginesie, o współpracy z zagranicznymi kolegami pomyśleli również młodzi informatycy z polskich klubów komputerowych?

Komentowana tu informacja z „Komsomolskiej Prawdy" ma jeszcze jeden aspekt. Otóż we wspólnym komunikacie ze spotkania na szczycie w Genewie pomiędzy Michaiłem Gorbaczowem i Ronaldem Reaganem znalazł się zapis stwierdzający, iż obaj przywódcy zgodzili się co do celowości współpracy w zakresie „opracowywania programów nauczania z pomocą mikrokomputerów dla szkół podstawowych i średnich". Z tego punktu widzenia to pierwsze spotkanie młodych informatyków amerykańskich i radzieckich jest jak gdyby cząstką realizacji tego ustalenia.

Genewskie spotkania na szczycie rozbudziło nadzieję na lepszą przyszłość świata. Spotkanie w Reykjawiku nadzieję tę jednak oddaliło. Groźny cień „gwiezdnych wojen" przeszkodził zawarciu porozumień, które dawałyby szansę likwidacji broni jądrowej do końca obecnego stulecia. Radziecki gest dobrej woli i najwyższej politycznej odpowiedzialności w postaci moratorium na wybuchy jądrowe zawisł w powietrzu i w momencie gdy piszę te słowa brak jest realnych przesłanek, by sądzić że do 1 stycznia 1987 roku Ameryka nie się do moratorium przyłączą. Dlatego, mówiąc w uproszczeniu dopóki, nie wróci zdrowy rozsądek politykom, o pokojową przyszłość świata muszą troszczyć się dzieci. Z Nowosybirska, z Chicago, z Krakowa ..

Nie trzeba jechać do Nowosybirska, aby przekonać się iż informatyka spowodowała, że świat stał się mały i bliski jak nigdy przedtem. Obowiązkiem informatyków, jak zresztą wszystkich odpowiedzialnie myślących ludzi na świecie, jest dbanie o to, aby ten świat w ogóle był.

Dosiego Roku! I wielu komputerów pod choinką!

Waldemar Siwiński





WYBIERZ SAM

„BAJTEK" — MIESIĘCZNY DODATEK DO „SZTANDARU MŁODYCH"

ADRES: 00-687 Warszawa, ul. Wspólna 61. Telefon 21-12-05

Przewodniczący Rady Redakcyjnej Jerzy Domański-redaktor naczelny „Sztandaru Młodych"

ZESPÓŁ REDAKCYJNY: Waldemar Siwiński (z-ca redaktora naczelnego „SM" — kierownik zespołu „Bajtka"), Roman Poznański (z-ca sekretarza redakcji „SM" — sekretarz zespołu „Bajtka"), Krzysztof Czernek, Sławomir Gajda (red. techniczny), Andrzej Kowalewski, Sławomir Polak, Wanda Roszkowska (opr. graficzne), Kazimierz Treger, Marcin Waligorski, Roman Wojciechowski

Klany redagują:
Commodore — Klaudiusz Dybowski, Michał Silski
Amstrad-Schneider — Tomasz Pyc, Sergiusz Wolicki
Spectrum — Konrad Fedyna, Michał Szuniewicz
Atari — Wiesław Migut

WYDAWCA RSW „Prasa-Książka-Ruch" Młodzieżowa Agencja Wydawnicza, al. Stanów Zjednoczonych 53, 04-028 Warszawa. Telefony: Centrala 13-20-40 do 49. Redakcja Reklamy 13-20-40 do 49 w. 403, 414

Cena 100 zł.

Skład techniką CRT-200, przygotowalnia offsetowa i druk: PRASOWE ZAKŁADY GRAFICZNE RSW „PRASA-KSIĄŻKA-RUCH" w Ciechanowie, ul. Sienkiewicza 51.

Zam. nr 1432/86, nakład 250 000 egz., P101

Bajtek



SZTUCZNA INTELIGENCJA: TO CO DZIŚ JEST TEORIĄ JUTRO STANIE SIĘ PRAKTYKĄ

Rozmowa z doc. dr hab. Adamem Borkowskim z pracowni Systemów Adaptacyjnych Ośrodka Rozwoju Techniki Instytutu Podstawowych Problemów Techniki PAN.

— Jules Verne najprostszy, kieszonkowy kalkulator uznałby za przejaw doskonałej sztucznej inteligencji. Dzis kalkulatory tak spowszedniały, że nie łączy się ich z tym pojęciem. Czy w ogole wiadomo, co to takiego sztuczna inteligencja?

— Pojęcie jest umowne, a roznica między tym, co inteligentne, a tym co nieinteligentne, polega głownie na sposobie opisu matematycznego. Za nieinteligentne uznajemy działania, które można w scisły sposob zalgorytmizowac opisac matematycznie za pomocą jakiegoś modelu. To, co poźniej wykonuje komputer, jest realizacją z gory dokładnie zaplanowanego trybu działan

— Nawet obliczanie liczby „pi"? Zadanie może wydac się niezbyt skomplikowane, a trwa swoisty wyscig, kto ułoży program na dokładniejsze wyznaczenie tej liczby. Rekordzistą jest David Bailey, ktory opracował algorytm na superkomputer Cray 2 i wyznaczył „pi" z dokładnością do 29 mln cyfr po przecinku. Japończycy odgrażają się, że najpierw dojdą do 33 mln, a poźniej aż do 100 mln cyfr.

— Ale nawet w tym wypadku komputer traktowany jest tradycyjnie, jako wzmacniacz możliwości człowieka. Właśnie jego tradycyjna rola polega na ułatwianiu działań na liczbach czy umożliwianiu szybkiego dostępu do dużych ilości informacji. Natomiast inteligentny komputer czy program wzmacnia nasze możliwosci w dziedzinie podejmowania decyzji oraz wnioskowania. I ta definicja wiąże się z potocznym rozumieniem inteligencji.

— Pod koniec lat pięćdziesiątych skonstruowano sztucznego szczura, ktory poruszał się w labiryncie, błądził, ale zapamiętywał przebytą drogę i powtornie nie popełniał błędow. W końcu znajdował wyjście. A więc elektroniczny szczur wnioskował i podejmował decyzje. Był w takim razie inteligentny?

— Według naszej definicji — tak. Dotychczas znane są automatyczne systemy, ktore pozwalają dojechac z punktu „a" do punktu „b", np. autopilot w samolocie czy urządzenia nawigacyjne na statku. Ale wszystkie te programy działają w sytuacji, gdy nie ma przeszkod, albo są one stałe lub usystematyzowane. Natomiast inteligentny [illegible] dobrze zachowac się w zmien[illegible] sytuacji, której nie można zawczasu prze[illegible]dziec, ktorej nie da się opisać za [illegible] ustalonego modelu matemat[illegible]

MYŚLĄCE ODKURZACZE

GRA O JUTRO

Jeśli przyjmiemy taką definicję sztucznej inteligencji, to możemy wiązać z nią następujące zagadnienia: systemy doradcze — programy typ ekspert, rozpoznawanie otoczenia za pomocą wizji, komunikacja z komputerem w języku naturalnym. Z tym ostatnim zagadnieniem wiążą się dwa problemy: rozpoznawanie przez komputer słów i zdań oraz generowanie mowy. Chyba najstarszą dziedziną sztucznej inteligencji są programy gier np. w szachy i tu są największe sukcesy. Natomiast najbardziej utylitarną dziedziną jest robotyka — inteligentne układy sterujące maszynami. Można jeszcze wyróżnić jako gałąź informatyki teorię sztucznej inteligencji. Spora grupa ludzi zajmuje się twierdzeniami, dowodami czy algorytmami podejmowania decyzji.

— Jaki jest stan badań nad sztuczną inteligencją? Czy wizje golemów i cyborgów nadal należą do pomysłów, które nie wykroczą poza literaturę science-fiction?

— Nie potrafię dać jednoznacznej odpowiedzi. Przede wszystkim badania nad sztuczną inteligencją są rozległe, interdyscyplinarne, a postęp jest zbyt szybki. To co jeszcze wczoraj było czystą teorią — dziś rozpatrywane jest w laboratoriach przez informatyków cybernetyków czy inżynierów-konstruktorów. A w ogóle sama nazwa „sztuczna inteligencja" jest prowokacyjna, często wywołuje niepotrzebne dyskusje i pytanie: czy można stworzyć golema, który będzie myślał sprawniej niż człowiek? To są rozważania ciekawe z filozoficznego punktu widzenia, lecz dziś ważniejsze wydają się konkretne problemy np komunikacji z komputerem. Są syntetyzery mowy, które wprawdzie dość nieprzyjemnie skrzeczą, ale można je jakoś zrozumieć. Gorzej jest z rozpoznawaniem naszej mowy przez komputery. Każdy naturalny język ma wieloznaczności, każdy człowiek mówi inaczej. Grozi niebezpieczeństwo, że sztuczny system źle rozpozna nasze rozkazy. Może to spowodować duże straty np. w sytuacji, gdy nie ma czasu na bieganie do pulpitu i naciskanie guzika. Maszyny reagujące na dziesięć słownych rozkazów typu „stój", „start" już są próbnie wprowadzane np. w kopalniach czy energetyce jądrowej.

— Niedawno firma Dragon System Inc. zademonstrowała prototypowy system umożliwiający prawidłowe rozpoznanie i reagowanie na ok. 2 tys. słów. Reakcja na jedno słowo poniżej jednej sekundy. Z kolei IBM opracowało program pozwalający uniknąć wieloznaczności takich słów jak „to", „too" czy „two", w zależności od kontekstu. Ale mimo zdaniem pojawia się wątpliwość: czy te starania nie wynikają bardziej z tęsknoty właśnie za człekopodobnym cyborgiem?

— Nie, takie rozumowanie byłoby dużym uproszczeniem, wręcz fantazjowaniem. Cechą rewolucji komputerowej jest dążenie do maksymalnego uproszczenia posługiwania się komputerami. Kiedyś polecenie dla komputera miało postać tasiemcowego wzoru, który nic nie znaczył dla przeciętnego człowieka. Dziś jeszcze nie marzymy, aby nauczyć komputer ludzkiego myślenia, ale dążymy do maksymalnego uproszczenia w posługiwaniu się nim. Sądzę, że w klasie komputerów personalnych czy domowych pojawią się niebawem takie, które będą rozumiały pewien określony zasób słów, np. rozkazy Basic'a, aby można było jakiś nieduży programik podać słownie.

— Jaka jest granica samodzielności komputera?

— Znów zahaczamy o zagadnienia natury filozoficznej i futurologicznej. Na razie nie ma odpowiedzi, a problem jest głównie rozważany przez ludzi zajmujących się teoretycznymi podstawami sztucznej inteligencji. Sprawa ma swoich gorących zwolenników jak i zimnych sceptyków, którzy uważają, że szkoda czasu na takie dylematy, a trzeba robić to, na co stać dzisiejszą teorię i technologię. Nie można w ścisły naukowy sposób dziś odpowiedzieć na pytanie: czy będziemy w stanie opracować program, który stworzy coś w rodzaju sztucznej świadomości? Czy program ten będzie na tyle sprawny w uczeniu się i wyciąganiu wniosków, ze stanie się jakby własnym ja? Czy jest to pryncypialnie możliwe? Zwolennicy jako dowód podają fakt, że możemy już dziś układać programy, które same się uczą. Tak było z programem uczącym się grać w warcaby. Po niedługim czasie wygrał on ze swoim twórcą.

— To już sukces!

— Tak, ale wynikający częściowo z tego, że zasady gra w warcaby są proste, potok informacji nie jest zbyt duży i wystarczyło dobrze podejść do sprawy. Przykład ten wskazuje, że wyróżnikiem sztucznej inteligencji jest ustawiczny kontakt z otoczeniem. Robota wyposaża się w zmysły — odpowiednie czujniki i na podstawie informacji płynących z czujników maszyna odtwarza sobie otoczenie, wyciąga wnioski i podejmuje decyzje. Największe osiągnięcia są w dziedzinie robotów przemysłowych, które potrafią np. z nieuporządkowanego stosu elementów wybrać właściwy i zamontować w odpowiednim miejscu powstającego urządzenia. Istnieją zautomatyzowane systemy transportu wewnątrzzakładowego, w których po hali fabrycznej poruszają się samodzielnie wózki, omijając przeszkody i docierając precyzyjnie do celu.

Coraz popularniejsze są systemy doradcze typu ekspert. Tu postęp jest szybki — pierwsze tego typu systemy powstały w medycynie do celów diagnostycznych. Dziś stosuje się je m.in. w doradztwie finansowym czy przy rozwiązywaniu problemów inżynierskich. Jako przykład można podać sprawdzanie stopnia bezpieczeństwa reaktora atomowego ze względu na przeciążenie. Jest to skomplikowane zagadnienie matematyczne i dobór właściwej metody wymaga często ukierunkowanej wiedzy. Nośnikiem takiej wiedzy może być program doradczy.

Przy pomocy systemów doradczych dokonuje się też diagnostyki skomplikowanych urządzeń. Mechanik naprawiający samoloty wystukuje na klawiaturze dane o awarii, a komputer daje wskazówki co mogło się zepsuć oraz jak należy usuwać usterkę. Niestety w innych dziedzinach nie ma jeszcze spektakularnych osiągnięć.

— Ale prognostycy przemysłu i handlu przewidują, że tak jak dziś jest popyt na urządzenia hi-fi, czy wideo — za kilkanaście lat będzie popyt na różnego rodzaju roboty kuchenne, domowe, do prac w ogródku itp.

— I jest to słuszna prognoza. Frajdą jest bowiem mieć w domu odkurzacz, który sam jeździ po pokoju, omija meble, czyści dokładnie dywan, bo mu się nie spieszy i nie ma problemu: lubi czy nie lubi sprzątać. A na dodatek gdy skończy — to sam się wyłącza. Takie urządzenia już są, a wszystko zależy od pomysłowości, która praktycznie jest nieograniczona.

— Co jest większą barierą w rozwoju sztucznej inteligencji: technologia czy informatyka?

— Sytuacja jest różna w różnych działach. Są dziedziny, gdzie koncepcyjnie sprawa jest rozwiązana, a o sukcesie decyduje szybkość przetwarzania informacji. Ale istnieją również działy, w których występują nadal duże trudności koncepcyjne, np jak zmagazynować wiedzę?

— Czy wierzy pan w stworzenie sztucznej inteligencji — według naszej definicji?

— Oczywiście, że wierzę! Można to uzasadniać w sposób filozoficzny: to co dziś jest teorią lub jest niemożliwe — jutro stanie się praktyką. Wspomniał pan o Vernie — dla niego kieszonkowy kalkulator byłby doskonałym, inteligentnym urządzeniem. Dziś aż tak wysoko kalkulatora nie cenimy. Dlaczego więc współczesne twory wyobraźni — oczywiście w rozsądnych granicach — nie mają zapełnic naszej planety za sto lat? W stworzeniu sztucznej inteligencji nie ma przeszkody natury zasadniczej — jest to kwestia czasu. Czy świadomość i zdolność myślenia jest zastrzeżona tylko dla ludzi? Niektórzy naukowcy całkiem poważnie zastanawiają się, czy oprócz naszego rozwoju węglowo-białkowego nie może istnieć rozwój krzemowy? Zgodnie z tą teorią, cyborgi można traktować jako zalążek nowej cywilizacji krzemowej.

Rozmawiał:
Roman Wojciechowski

MIKROREWOLUCJA

Prezentując typowe zastosowania mikrokomputerów prędzej czy później musimy trafic do szkoły. Nie po to, żeby znów dyskutować o tym jak uczyć informatyki. Przed ekranem interesują nas głównie korzyści, jakie może miec z komputera człowiek, który umie tylko włączyć go do sieci i uruchomić zapisany na taśmie czy dyskietce program.

Inaczej mówiąc, dziś traktujemy nasze mikro nie jako przedmiot nauczania, ale jako pomoc naukową, która ma wspierać w pracy nauczyciela i uczniów.

Żeby była to pomoc naprawdę skuteczna a nie kosztowna, modna zabawka potrzebne są dobre programy. Programy przeznaczone do nauki różnych przedmiotów i dla różnych wiekiem grup uczniów. Jednym słowem, potrzeba dużo programów, a im jest ich więcej tym lepiej, po to żeby każdy nauczyciel miał z czego wybrać to, co dla jego wychowanków będzie najlepsze.

Ci z Was, którzy chodzą w tej chwili do szkoły zdają sobie sprawę, że na razie nasze szkolnictwo nie dostarczy mi wiele materiału na artykuły o programach edukacyjnych. Spróbowałem więc poszukać przykładów trochę dalej. Ponieważ jednak na świecie takich programów jest dużo więcej niż można (i niż warto) tu zaprezentować, musimy się zdecydować na coś konkretnego.

Podstawowym językiem informatyki nie jest wcale BASIC czy PASCAL, lecz język angielski, więc może przyjrzyjmy się kilku programom przeznaczonym do nauki tego właśnie języka. Niestety wszystkie, o których będzie mowa, przeznaczone są na mikrokomputer o nazwie BBC, bardzo popularny w Wielkiej Brytanii, lecz u nas prawie nie występujący.

W oryginalnym wydaniu każdy z programów zapisany jest na oddzielnej dyskietce co ułatwia jego uruchamianie. Wkładamy dyskietkę do stacji, naciskamy dwa (dla wszystkich programów te same) klawisze funkcyjne i program ładuje się do pamięci oraz startuje samoczynnie. Wszystkie programy zaczynają pracę od wyświetlenia krótkiej informacji o swoich możliwościach i o sposobach ich wykorzystania. Wszystkie są sterowane przez menu, czyli wyświetlają na ekranie listę dostępnych możliwości do wyboru dla użytkownika. Atrakcyjna, ko-



[illegible] i układy scalone



Przyrządy optoelektroniczne

lorowa szata graficzna uprzyjemnia pracę ale pełni również funkcje praktyczne — kolory są wykorzystywane np. do dodatkowego rozróżniania informacji różnego typu. Często spotykana jest opcja pozwalająca przeliczać osiągnięcia ucznia na punkty i zliczać wyniki. Pozwala to śledzić postępy, lub uatrakcyjniać naukę przez organizację zawodów w kilkuosobowych grupach korzystających z tego samego programu.

Po tych ogólnych spostrzeżeniach przejdźmy do opisu kilku spośród wielu dostępnych na BBC programów. Zacznijmy od gry **„London Adventure"**. Zasady są bardzo proste: wracasz do domu z wakacji w Londynie, spieszysz się żeby zdążyć na samolot a jeszcze musisz kupić prezenty w sklepach rozrzuconych po całym mieście. Możesz korzystać z metra lub taksówki ale pieniędzy też nie masz za wiele. Gra odbywa się w czasie zbliżonym do rzeczywistego, i chociaż nie ma w niej jadowitych pająków ani tajemniczych szkieletów, może trochę wciągnąć. A wtedy, prawie nieświadomie, przyswajasz i ćwiczysz całe serie zwrotów potrzebnych w codziennym życiu — pytania o drogę, określanie miejsca, rozmowy w sklepie, kupno biletów itd. Musisz to wszystko robić poprawnie, bo jeśli zwracasz się błędnie lub niegrzecznie, to Twoi rozmówcy odchodzą bez słowa, a godzina odlotu zbliża się przecież nieubłaganie.

Po grze kolej na ćwiczenia stylistyczne z programem **„Varietext"** Wszyscy wiemy, że w określonym kontekście pewne słowa pasują a ich synonimy nie bardzo. Ćwiczenie polega na uzupełnianiu tekstu słowami wybranymi spośród kilku bliskoznacznych propozycji. Na zakończenie możemy porównać wynik z tekstem źródłowym.

Następny program to **„Besieged"**, czyli „Oblężony". Oblężony jest zamek Kraaka, a Ty jesteś rycerzem, który spieszy z odsieczą. Jednak żeby dostać się do środka musisz pokonać fosę. Program wyświetla definicję słowa, które masz odgadnąć. Prawidłowa odpowiedź buduje most, po którym wjeżdżasz do zamku. Jeśli nie jesteś pewien możesz próbować zgadywać, ale nieprawidłowo podane litery wpadają do fosy i, gdy ją zasypią, do zamku wdzierają się wrogowie. Jeśli masz trudności możesz prosić o dodatkową wskazówkę — informację o literach, z których składa się odpowiedź.

Nietrudno zgadnąć, że jest to ubrane w atrakcyjną szatę ćwiczenie wzbogacające słownik. Podobne zadanie spełnia program o nazwie **„Wordpower"**, w którym trzeba łączyć w pary wyrazy powiązane ze sobą np. synonimy, słowa o przeciwnym znaczeniu, czy rzeczowniki z przymiotnikami, które od nich pochodzą. „Wordpower" przewidziany jest dla uczniów od dziesięciu lat wzwyż, więc też ma formę zabawy. Z zadanego słowa trzeba strzelać tak by trafić we właściwą odpowiedź.

Skoro jesteśmy przy zabawach, to może wolicie **„Code Breaker"** czyli coś w rodzaju językowego Master Minda. Prezentuje poprawne angielskie zdanie, zakodowane w ten sposób, że każda litera jest zastąpiona inną (w całym zdaniu tą samą). Trzeba złamać kod, a przy okazji, niejako mimochodem, zapoznajmy się z angielskimi słowami i zdaniami — zestaw zdań na dyskietce jest bogaty i tak przygotowany, aby korzyści językowe były największe. Jeśli chcemy użyć programu jako gry towarzyskiej, to można również podawać z klawiatury zdania, które program „od ręki" zakoduje i przedstawi do rozwiązania przeciwnikowi.

I na koniec coś poważniejszego **„Screentest for First Certificate"**. Duży program, pozwalający sprawdzać znajomość angielskiego na poziomie wymaganych do uzyskania podstawowego świadectwa znajomości języka, wydawanego cudzoziemcom. Może być również używany jako zbiór ćwiczeń w zakresie podstawowych konstrukcji językowych. Podzielony jest na pięć tematycznych działów, z których każdy zawiera cztery różniące się stopniem trudności zadania. Każde z zadań można wykonywać na dwa sposoby. W pierwszym z nich, dla każdego z pytań, komputer proponuje kilka odpowiedzi, spośród których trzeba wybrać właściwą. W drugim, pytania są te same, ale odpowiedź trzeba wpisać samemu. Program na bieżąco ocenia odpowiedzi a także wyświetla inne prawidłowe możliwości oraz krótki komentarz. Po wykonaniu wszystkich pozycji jednego zadania wystawiana jest za nie ocena ostateczna, która jest również zapisywana na dyskietce przy nazwisku ucznia. Daje to nauczycielowi możliwość sprawdzenia wyników.

Statystyka wyników prowadzona dla poszczególnych zadań pozwala łatwo ustalić co sprawia uczniom największą trudność. Nauczyciel może również zaprogramować kolejność w jakiej komputer będzie przedstawiał zadania do rozwiązania, czy też zmieniać ilość punktów wymaganą na zaliczenie. Program jest opracowany bardzo starannie — przewidziano nawet przetasowywanie proponowanych do wyboru odpowiedzi, żeby przy kilkakrotnym wykonywaniu tego samego ćwiczenia wyeliminować podświadome zapamiętywanie pozycji prawidłowego wariantu zamiast jego treści.

Myślę, że powyższa próbka wystarczy do wyobrażenia sobie innych możliwych programów edukacyjnych. Być może niektórzy z Was są trochę rozczarowani — „Przecież takie programy nie zastąpią nauczyciela". Oczywiście, że nie zastąpią go całkowicie, ale wyręczą go w wielu żmudnych funkcjach, np. w wielokrotnym wykonywaniu z uczniami tych samych ćwiczeń. Poza tym pozwalają na indywidualną pracę ucznia z tymi, co akurat jemu sprawia szczególne trudności, lub najbardziej go interesuje. Pewne części materiału mogą być opanowywane czy utrwalane bez udziału nauczyciela, nawet w domu.

Ogromne znaczenie ma również fakt, że zastosowanie komputerów uatrakcyjnia pracę, pozwala uczyć się podczas zabawy, zwiększa zainteresowanie nauką, a przez to wszystko zapobiega nudzie, która jest chyba najgorszym wrogiem dobrych wyników.

Andrzej Pilaszek



NAJTRUDNIEJSZY ZAWÓD

Mikrokomputer ośmiobitowy, który stawiamy na biurku, wykonany techniką lampową w latach czterdziestych byłby wielkości czteropiętrowego domu. Ten sam mikrokomputer na tranzystorach z trudem mieściłby się w M-5. A dziś już przecież konstruuje się komputery jednoukładowe wielkości pudełka do zapałek. Tę rewolucję umożliwiło wprowadzenie do produkcji układów scalonych wielkiej skali integracji — VLSI (Very Large Scale of Integration).

Zamiast wykonywać na płytce krzemowej jeden tranzystor umieszcza się ich nawet kilkaset tysięcy razem z rezystorami, diodami i kondensatorami małej pojemności. Oczywiście wymaga to wielkiej precyzji i bardzo dokładnego przestrzegania reżimów technologicznych. W największym skrócie wygląda to następująco: najpierw trzeba uzyskać monokryształ krzemu, pociąć go na płytki grubości od kilku do kilkudziesięciu mikronów. Potem na tych płytkach idealnie wypolerowanych wytrawia się i napyla kolejne warstwy półprzewodników. O skali tych operacji świadczy najlepiej to, że przebiegają one w substancjach niemal wielkości atomów. Gotowe płytki o rozmiarach 5×5 (a nawet mniej) umieszcza się na tzw. strukturze nośnej. Teraz pozostaje tylko przylutowanie wyprowadzeń złotym drutem kilkaset razy cieńszym od włosa i ... umieszczenie wszystkiego w obudowie plastikowej lub ceramicznej. Skonstruowany w 1982 układ scalony zawierał 450 tys. tranzystorów.

Prekursorem elektroniki półprzewodników w świecie był m.in. prof. Cezary Andrzej Ambroziak, który uczestniczył w opracowaniu pierwszych technologii układów scalonych w USA. W Polsce pierwsze prace w zakresie elektroniki półprzewodników rozpoczął na przełomie lat 1952–1953 zespół kierowany przez prof. Janusza Groszkowskiego, a więc w pięć lat po wynalezieniu tranzystora. Wkrótce rozpoczęto doświadczalną produkcję elementów półprzewodnikowych. W roku 1958 utworzono Fabrykę Tranzystorów TEWA i Zakład Produkcji Półprzewodników PEWA. Od tych lat datuje się rozwój przemysłu półprzewodnikowego.

Obecnie w kraju jedynym producentem półprzewodników jest Naukowo-Produkcyjne Centrum Półprzewodników UNITRA — CEMI, które produkuje ok. 30 mln. układów scalonych średniej i dużej skali integracji bipolarnych i unipolarnych — w tym rodzinę układów 8-bitowego mikroprocesora MCY 7880 (odpowiednik INTEL 8080). Oprócz tego CEMI produkuje elementy dyskretne jak. diody — począwszy od prostowniczych po stabilizacyjne, tranzystory wszystkich rodzajów oraz elementy optoelektroniczne (diody elektroluminescencyjne, fototranzystory i tranzystory). CEMI współpracuje z producentami w krajach socjalistycznych zarówno w opracowaniu nowych generacji układów jak i w specjalistycznej produkcji.

Jakie są perspektywy produkcji? Odpowiedź na to pytanie mgr inż. Jerzy Bujok — zastępca d.s. technicznych kierownika Zakładu Produkcji Układów Scalonych — zaczyna od wyjaśnienia na czym polega reżim technologiczny. Na przykład poziom czystości w hali, gdzie powstają VLSI mierzony jest ilością pyłków w stopie sześciennej (obowiązują angielskie miary). Nie może być ich więcej nić dziesięc — każdy o rozmiarach do pół mikrona! Przekładając te dane na prostszy język — jeden pyłek na dcm sześć. Dla porównania — w powietrzu ulicznym średnio w 1 dcm^3 znajduje się 100 tys. pyłków. Natomiast dokładność temperatury w procesach technologicznych określa się do połowy stopnia Celsjusza przy tysiącu stopni — czyli jest to dokładność półpromilowa. Ponieważ produkcja układów scalonych to ingerencja na poziomie atomów — wymagana jest wysoka klasa stabilności i powtarzalności technologicznych. Przy takich wymaganiach niebagatelne znaczenie ma nawet rodzaj farby do malowania ścian w halach.

Rozwój elektroniki uwarunkowany jest poziomem technologicznym całej gospodarki kraju — mówi mgr inż. Jerzy Bujok. W tej chwili na świecie układy scalone projektowane są przy udziale komputerów. Również produkcja jest w pełni zautomatyzowana i kierowana przez komputery.

Obecnie możliwości produkcyjne CEMI zaspokajają połowę zapotrzebowania w kraju na elementy półprzewodnikowe i mniej niż jedną czwartą na mikroprocesory. Największymi barierami są ograniczenia inwestycyjne oraz brak wysoko specjalizowanych linii produkcyjnych. Linie te produkowane są przez nieliczne firmy zachodnie, a na ich sprzedaż do krajów socjalistycznych nałożone jest embargo. Ograniczenia te zmuszają do korzystania z własnej bazy naukowo-technicznej. Szansą na przełamanie impasu jest ściślejsza współpraca między krajami socjalistycznymi w dziedzinie rozwoju nowych technologii.

Najbliższe plany CEMI przewidują m.in., że do 1990 roku będzie seryjnie produkowana rodzina procesorów kompatybilnych z rodziną INTEL-a oraz pamięci dynamiczne i stałe do 16 kilobitów.

Sławomir Polak
Roman Wojciechowski

WSTĘP DO PROGRAMOWANIA W JĘZYKU

PASCAL

cz. III

Do tej pory posługiwaliśmy się w PASCAL-u jedynie tzw. typami prostymi danych; znamy ich jak dotąd cztery:

BOOLEAN INTEGER
CHAR REAL

ZŁOŻONE STRUKTURY DANYCH

Zapoznamy się obecnie z kilkoma dostępnymi w PASCAL-u typami strukturalnymi: tablicami, rekordami i zbiorami. Pojęcie tablicy było już na łamach „Bajtka" wyjaśniane (nr 7/86, str 30), dlatego też nie będziemy go tutaj przybliżać. W PASCAL-u mamy możliwość deklarowania i używania tablic o dowolnej liczbie wymiarów.

Zmienną Tablica, stanowiącą jednowymiarową, 10-elementową tablicę liczb całkowitych możemy zadeklarować np. jako:

var Tablica : array [1..10] of integer;

Podobnie tablice o większej ilości wymiarów mogą być deklarowane w sposób następujący:

var Szachownica : array [1..8, 1..8] of integer;
StronaTekstu : array [0..31, 0..63] of char;
Rubik : array [-1..1, -1..1, -1..1] of boolean;

Zauważmy, że możemy dowolnie wybrać typ elementów tablicy. Możliwe jest na przykład zadeklarowanie tablicy tablic (w istocie tak właśnie tworzone są tablice wielowymiarowe):

var Iks: array [1..100] of array [1..3] of boolean;

Mamy także możliwość dowolnego dobierania zakresów indeksów tablicy. Co więcej, w celu identyfikacji elementu tablicy możemy użyć nie tylko liczb całkowitych, ale np. typu CHAR

var Szachownica : array ['A'..'H', 1..8] of integer;

Przypuśćmy, że chcemy teraz nadać wszystkim elementom tablicy Szachownica wartość 0. Należy wówczas wykonać poniższy fragment programu:

for Znak := 'A' to 'Z' do
for Liczba := 1 to 8 do
Szachownica [Znak, Liczba] := 0;

gdzie zmienna Znak jest typu CHAR, zaś zmienna Liczba — typu integer. Przykład ten pokazuje, w jaki sposób odwołujemy się w programie do określonego elementu tablicy.

Można uczynić spostrzeżenie, że tablice, których elementami są znaki umożliwiają w PASCAL-u przechowywanie ciągów znaków jako wartości zmiennych. Dotychczas bowiem mogliśmy nadać zmiennej wartość wyłącznie pojedynczego znaku.

const DlSlowa = 12;
var Slowo : array [1..DlSlowa] of char;

Poniższy program ilustruje użycie takiej tablicy. Program wczytuje podany wyraz i drukuje go „od tyłu" *)

```
program Odwrocenie;

{ Program wczytuje do tablicy znakowej Wyraz
  podane z klawiatury slowo, a nastepnie
  wypisuje je w odwrotnej kolejnosci. }

const DlSlowa = 20;

var Wyraz    : array [1..DlSlowa] of char;
    Indeks,k : integer;
    Znak     : char;

begin
{ Wczytanie wyrazu }
  writeln (' Podaj wyraz (TYLKO DUZE LITERY): ');
  Indeks := 1;
  read (Znak);
  while (Znak >= 'A')
        and (Znak <= 'Z')
            and (Indeks <= DlSlowa) do
    begin
      Wyraz [Indeks] := Znak;
      Indeks := Indeks + 1;
      read (Znak)
    end;
 { Wypisanie wyrazu odwroconego }
  writeln;
  for k := Indeks downto 1 do
    write (Wyraz [k]);
  writeln;
 end.
```

Standardowa wersja PASCAL-a nie dopuszcza stosowania ciągów znaków o zmiennej długości — co stanowi pewną niedogodność (np. w powyższym przykładzie zbędne, nie będące częścią wyrazu, elementy tablicy Slowo musieliśmy zastąpić spacjami w celu ich wyróżnienia). W nowszych implementacjach wprowadzony został typ standardowy STRING, którym posługujemy się podobnie, jak typem łańcuchowym w BASIC-u. W celu szczegółowego zapoznania się z możliwością użycia tego typu należy jednak sięgnąć do opisu wersji PASCAL-a, którą się aktualnie posługujemy.

Przykładem użycia tablic znakowych może być również poniższy program, szyfrujący podany fragment tekstu według uprzednio podanego klucza. Kluczem jest tutaj ciąg 26 znaków, odpowiadających w naszym kodzie kolejnym literom alfabetu *)

```
program Szyfr;

{ Program szyfrujacy podany ciag znakow wedlug
  podanego wczesniej klucza 26-znakowego }

var Klucz     : array ['A'..'Z'] of char;
    Tekst     : array   [1..40] of char;
    Indeks, k : integer;
    Znak      : char;

begin
{ Wczytanie klucza szyfru }
  writeln (' Podaj 26-znakowy klucz ');
  for Znak := 'A' to 'Z' do
    read (Klucz [Znak]);
 { Wczytanie jednego wiersza tekstu }
  Indeks := 1;
  while (Znak <> chr (13)) and (Indeks <= 40) do
    begin
      read (Tekst [Indeks]);
      Indeks := Indeks + 1
    end;
{ Wypisanie tekstu zaszyfrowanego }
  writeln;
  for k := 1 to Indeks - 1 do
    if Tekst [k] = ' ' then
      write (' ')
    else
      write (Klucz [Tekst [k]]);
end.
```

W tablicy wszystkie elementy muszą być tego samego, z góry określonego, typu. W wielu sytuacjach nie jest to wygodne — np. gdy zachodzi potrzeba przechowywania w pamięci informacji różnego typu, ale dotyczących jednego obiektu. Możemy np. zażądać utworzenia kartoteki, w której każda „strona" zawierać będzie nazwisko i imię oraz numer telefonu pewnej osoby, przy czym nazwisko i imię chcemy reprezentować w postaci ciągu znaków, zaś numer telefonu — jako liczbę naturalną. „Strony" takiej kartoteki nie możemy wówczas reprezentować w postaci jednej tablicy — sensownym staje się utworzenie innej złożonej struktury danych, w której każdy element mógłby być innego typu. Strukturą taką jest rekord. Rekord złożony jest z pewnej ilości pól, identyfikowanych nie przez indeks, jak w przypadku tablic, lecz przez

nazwę. Każde pole rekordu ma osobno zdefiniowany typ.

Definicja zmiennej rekordowej odpowiadającej wspomnianej „stronie" kartoteki może mieć postać następującą:

```
var Strona : record
              Imie : array [1..12] of char;
              Nazwisko : array [1..12] of
               char;
              Telefon : 0..999999;
             end;
```

Po słowie RECORD następuje lista nazw pól rekordu wraz z ich typami. Definicję rekordu kończymy słowem END.

Do poszczególnych pól rekordu Strona odwołać się w programie możemy np. następująco:

```
Strona.Telefon := 211205;
Strona.Imie [1] := 'B';
for k := 1 to 12 do
  read (Strona. Nazwisko [k]);
```

Zapis tych kilku instrukcji jest równoważny nieco innemu:

```
with Strona do
  begin
    Telefon := 211205;
    Imie [1] := 'B';
    for k := 1 to 12 do
    read (Nazwisko [k])
end;
```

Użyta tutaj instrukcja WITH może być interpretowana jako „z rekordem Strona wykonaj:". Po jej użyciu nie jest już potrzebne podawanie nazwy rekordu przy jej odwoływaniu się do jego pól.

Naszą kartotekę, liczącą, powiedzmy, 100 stron możemy teraz zdefiniować jako tablicę rekordów.

```
var Kartoteka : array [1..100] of
                record
                  Imie : array [1..12] of char;
                  Nazwisko : array [1..12] of
                   char;
                  Telefon : 0..999999;
                end:
```

Odwołanie się do np. telefonu na stronie 1 kartoteki wymaga teraz dłuższego zapisu:

```
Kartoteka [1].Telefon := 211205;
```

albo

```
with Kartoteka [1] do
  Telefon := 211205;
```

Rekordy są w wielu wypadkach bardzo użytecznymi strukturami danych. Między innymi pozwalają one na konstrukcję w PASCAL-u struktur listowych. Ze względu na elementarny charakter niniejszego wykładu zagadnienia tego nie omówimy, odsyłając czytelnika do literatury.

PASCAL przewiduje również możliwość wykonywania działań na zbiorach. Oto deklaracja zmiennej Alfa jako zbioru.

```
var Alfa : set of integer;
```

Elementy zbioru muszą być typu prostego — zatem nie jest możliwe np. utworzenie zbioru tablic. Należy też pamiętać, że ilość elementów zbioru jest ograniczona. W niektórych wersjach PASCAL-a zbiory mogą zawierać tylko do 16 elementów; z reguły jednak ograniczenie to jest równe przynajmniej 64.

Działając na zbiorach możemy używać działań i relacji znanych z teorii mnogości. Oto kilka przykładów.

```
Alfa :=[ ];
```

Zmienna Alfa przyjmuje wartość zbioru pustego.

```
Alfa := Alfa + [0, 1, 2, 3, 4];
```

Zmienna Alfa przyjmuje wartość złączenia (sumy mnogościowej) zbiorów Alfa i [1, 2, 3, 4]. Należy zwrócić uwagę, że znak + użyty tutaj nie oznacza dodawania

```
Alfa := Alfa * [0, 3, 7];
```

Znak * jest tutaj operatorem przecięcia (iloczynu zbiorów), nie zaś mnożenia.

Mamy też możliwość stwierdzenia, czy dany element X należy do zbioru Alfa.

```
if X in Alfa then
  writeln ('Nalezy')
else
  writeln ('Nie nalezy');
```

Oprócz powyższych, możemy używać jeszcze operatorów relacji równości, nierówności oraz inkluzji (zawierania się).

A = B — true, jeżeli zbiory A i B są równe, np. [1, 2] = [2, 1]

A <> B — true, jeżeli zbiory A i B są różne.

A < =B — true, jeżeli zbiór A zawiera się w B.

A > =B — true, jeżeli zbiór B zawiera się w A.

Zauważmy, że znaczenie używanych tu symboli relacji jest inne od przyjętego przy porównywaniu np liczb.

DEFINIOWANIE TYPÓW DANYCH

Podana nieco powyżej deklaracja zmiennej Kartoteka zajmuje dość sporo miejsca, ze względu na rozbudowaną definicję jej typu. Deklarację tą możemy uprościć, jeżeli uprzednio nadamy używanym typom zmiennych własne nazwy. Dokonuje się tego w części programu definiującej typy, umieszczonej zaś bezpośrednio po definicji stałych w programie — a zatem, zgodnie z zasadą predefinicji, przed deklaracjami zmiennych.

Zmodyfikowana część programu deklarująca zmienną Kartoteka może mieć postać następującą

```
const DlWyrazu = 12;
      IloscStron = 100;
type Wyraz = array [1..DlWyrazu] of char;
     Strona = record
               Imie : Wyraz;
               Nazwisko : Wyraz;
               Telefon : 0..999999;
              end;
var Kartoteka : array [1..IloscStron] of Strona;
```

Zauważmy, że przy konieczności wielokrotnego deklarowania zmiennych typu Wyraz lub Strona zysk na objętości i przejrzystości programu jest znaczny.

Skrócenie zapisu nie jest jednak jedynym powodem, dla którego PASCAL został wyposażony w możliwość definiowania typów danych. Programista może bowiem określić swoje własne typy, zależnie od potrzeb danego programu. Są to tzw. typy wyliczeniowe.

```
type DzienTyg = (Pon, Wto, Sro, Czw, Pia,
      Sob, Nie);
     DzienRoboczy = Pon..Pia;
```

W powyższym przykładzie definiujemy typ DzienTyg poprzez wymienienie (w porządku rosnącym) wszystkich wartości, jakie może przyjmować zmienna tego typu. Typ DzienRoboczy powstał przez okrojenie zdefiniowanego wcześniej typu wyliczeniowego.

Ze względu na określony w definicji typu porządek możemy w odniesieniu do zmiennych i wyrażeń typu wyliczeniowego używać następujących relacji i funkcji

= <> < > <= >=

— jak w odniesieniu do innych typów prostych

succ (x) — element następny względem x. Należy pamiętać, że największy element typu nie ma następnika.

pred (x) — element poprzedni względem x. Najmniejszy element typu nie ma poprzednika.

Typy wyliczeniowe mogą służyć do indeksowania tablic oraz kontrolowania pętli FOR. Wartości tych typów nie mogą być natomiast wypisywane przez WRITE.

Marek Wyrwidąb

*) Dwa podane w tekście artykułu programy napisane zostały w zgodzie z raportem Pascala. W wersjach języka, używających innego aparatu wykonywania operacji wejścia, programy te są poprawne składniowo, lecz mogą pojawić się problemy przy wczytywaniu danych

```
program Odwracanka;

{ Gra w przewracanke.  Dana jest kwadratowa plansza
  o wymiarach  5 x 5,  ktorej  wszystkie  pola  sa
  "ciemne".  Gra  polega na  "zapaleniu" wszystkich
  pol planszy. Ruch polega na zmianie stanu jednego
  z pol, przy czym  zmianie ulegaja  rowniez cztery
  pola sasiednie. }

type Szachownica = array [1..5, 1..5] of boolean;

var Plansza : Szachownica;
    i, j    : integer;
    Koniec  : boolean;

begin
{ Poczatkowy stan planszy }
  for j := 1 to 5 do
    for i := 1 to 5 do
      Plansza [i, j] := false;
  Koniec := false;
  while not Koniec do
    begin
{ Drukowanie planszy }
      writeln;
      writeln (' +---+---+---+---+---+ ');
      for i := 1 to 5 do
        begin
          write (' ! ');
          for j := 1 to 5 do
            if not Plansza [i, j] then
              write ('X ! ')
            else
              write ('  ! ');
          writeln;
          writeln (' +---+---+---+---+---+ ');
        end;
{ Wczytanie wspolrzednych ruchu }
      writeln (' PODAJ    WSPOLRZEDNE    RUCHU ');
      writeln (' (wiersz, kolumna - 1..5, 1..5):');
      read (i, j);
{ Wykonanie ruchu }
      Plansza [i, j] := not Plansza [i, j];
      if j + 1 < 6 then
        Plansza [i, j + 1] := not Plansza [i, j + 1];
      if j - 1 > 0 then
        Plansza [i, j - 1] := not Plansza [i, j - 1];
      if i + 1 < 6 then
        Plansza [i + 1, j] := not Plansza [i + 1, j];
      if i - 1 > 0 then
        Plansza [i - 1, j] := not Plansza [i - 1, j];
{ Sprawdzenie, czy zachodzi koniec gry }
      Koniec := true;
      for i := 1 to 5 do
        for j := 1 to 5 do
          if not Plansza [i, j] then
            Koniec := false;
    end;
  writeln;
  writeln (' GRATULACJE - WYGRALES !!! ');
end.
```

czar CZTERECH KÓŁEK

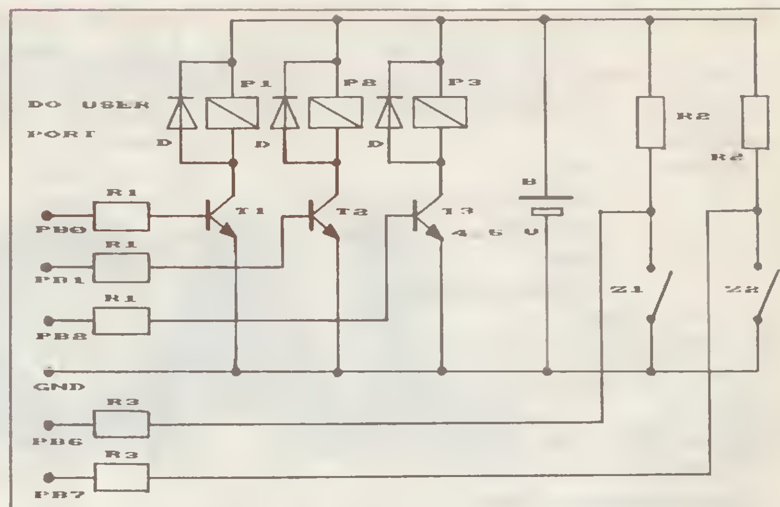
Postanowiliśmy, że w dwunastym, gwiazdkowym numerze „Bajtka" musi znaleźć się coś specjalnego. Proponujemy więc Wam skonstruowanie samochodziku sterowanego przez komputer, a dla mniej zaawansowanych w pracach ręcznych — układu oświetlenia choinki.

Oba projekty wymagają wiedzy o elektronice, trochę czasu i odrobinę odwagi (złe podłączenie przewodów do gniazda USER PORT może spowodować uszkodzenie niektórych obwodów wewnątrz komputera, a w konsekwencji niemożliwość używania takich urządzeń jak szybka stacja dyskow czy modem).

Do skonstruowania zdalnie sterowanego samochodziku potrzebny jest sześciożyłowy przewód, trzy kontaktronowe przekaźniki o jak najmniejszym poborze prądu, elementy elektroniczne jak oporniki i diody — podane w spisie, wtyk do USER PORTU (dwustronne złącze krawędziowe o odpowiednio dobranym rozstawie styków), no i oczywiście samochodzik.

Samochodzik musi być wyposażony w dwa silniczki 4.5 V, jeden napędzający lewe, drugi prawe koło. Przyjęcie takiego rozwiązania powoduje, że jeśli obracają się oba silniki, samochód jedzie do przodu lub do tyłu, jeśli natomiast działa tylko jeden silnik, samochód skręca.

Warto też postarać się o to, by nasz samochodzik mógł wykrywać zderzenia. W tym celu z przodu i z tyłu nadwozia montujemy po dwie sprężynujące blaszki. Normalnie są one rozwarte, podczas zderzenia stykają się. Z1 to zderzak przedni, Z2 — tylni.



Rys. 1 Schemat układu sterującego samochodzikiem.

Rys. 1. przedstawia schemat układu sterowania samochodzikiem. Układ ten można umieścić wewnątrz karoserii modelu. Z nadwozia zabawki wyprowadzamy 6 przewodów połączonych ze stykami gniazda USER PORT.

Układ jest zasilany z baterii 4.5 V. Podczas montażu należy zwrócić szczególną uwagę na prawidłowość połączeń — wspominałem już o grożących niebezpieczeństwach.

Połączenie styków przekaźników z silnikami pokazane jest na rys. 2. Tym, którzy

USER PORT — widok od strony wtyczki:

Commodore 64

1 2 3 4 5 6 7 8 9 10 11 12

A B C D E F H J K L M N

PIN	OZNACZENIE	PIN	OZNACZENIE
1	GND	A	GND
2	+5V, max. 100 mA	B	CB 1
3	RESET	C	PB 0
4	JOY 0	D	PB 1
5	JOY 1	E	PB 2
6	JOY 2	F	PB 3
7	LIGHT PEN	H	PB 4
8	CASSETTE SWITCH	J	PB 5
9	SER. ATN IN	K	PB 6
10	9V AC, max. 100 mA	L	PB 7
11	9V AC, max. 100 mA	M	CB 2
12	GND	N	GND

VC20

1 2 3 4 5 6 7 8 9 10 11 12

A B C D E F H J K L M N

PIN	OZNACZENIE	PIN	OZNACZENIE
1	GND	A	GND
2	+5V, max. 100 mA	B	FLAG 2
3	RESET	C	PB 0
4	CNT 1	D	PB 1
5	SP 1	E	PB 2
6	CNT 2	F	PB 3
7	SP 2	H	PB 4
8	PC 2	J	PB 5
9	SER. ATN IN	K	PB 6
10	9V AC, max. 100 mA	L	PB 7
11	9V AC, max. 100 mA	M	PA 2
12	GND	N	GND

Program obsługi samochodu

```
10 REM PROGRAM OBSŁUGI SAMOCHO.
DU   ZABAWKI
20 PRINTCHR$(147):D$=CHR$(17)
25 PRINT" OZNACZENIE KLAWISZY
"
30 T$(1)="-N- RUCH DO PRZODU"
:PRINTD$,T$(1)
35 T$(2)="-X- RUCH DO TYŁU":PR
INTD$,T$(2)
40 T$(3)="-M- RUCH W PRAWO - D
O PRZODU":PRINTD$,T$(3)
45 T$(4)="-B- RUCH W LEWO - DO
 PRZODU":PRINTD$;T$(4)
50 T$(5)="-C- RUCH W PRAWO - D
O TYŁU":PRINTD$;T$(5)
55 T$(6)="-Z- RUCH W LEWO - DO
 TYŁU":PRINTD$;T$(6)
60 T$(7)="-V- STOP":PRINTD$;T$
(7)
65 PRINTD$,D$:D$
70 FORA=1TO10:BL$=BL$+CHR$(32)
:NEXT
100 K=56579:B=56577
110 POKEK,7:POKEB,0
500 GETX$
510 IFX$="N"THENGOSUB1000
520 IFX$="X"THENGOSUB1030
530 IFX$="M"THENGOSUB1060
540 IFX$="B"THENGOSUB1090
550 IFX$="C"THENGOSUB1120
560 IFX$="Z"THENGOSUB1150
570 IFX$="V"THENGOSUB1180
580 GOSUB2000
590 GOTO500
1000 REM DO PRZODU
1010 PRINTCHR$(145);T$(1);BL$
1020 POKEB,6:RETURN
1030 REM DO TYŁU
1040 PRINTCHR$(145);T$(2);BL$
1050 POKEB,7:RETURN
1060 REM W PRAWO DO PRZODU
1070 PRINTCHR$(145);T$(3);BL$
1080 POKEB,2:RETURN
1090 REM W LEWO DO PRZODU
1100 PRINTCHR$(145),T$(4);BL$
1110 POKEB,4:RETURN
1120 REM W PRAWO DO TYŁU
1130 PRINTCHR$(145);T$(5),BL$
1140 POKEB,3:RETURN
1150 REM W LEWO DO TYŁU
1160 PRINTCHR$(145);T$(6);BL$
1170 POKEB,5:RETURN
1180 REM * STOP *
1190 PRINTCHR$(145),T$(7);BL$,
BL$
1200 POKEB,0:RETURN
2000 REM * ZDERZENIE *
2010 ZD=PEEK(B)AND192
2020 IFZD=192THENRETURN
2030 IFZD=64THENPRINTCHR$(145)
"ZDERZENIE";BL$,BL$:GOTO1170
2040 IFZD=128THENPRINTCHR$(145
)"ZDERZENIE";BL$;BL$:GOTO1080
2050 PRINTCHR$(145)"DWA ZDERZE
NIA NA RAZ ?!":GOTO1200

READY.
```

Program sterowania oświetleniem choinkowym

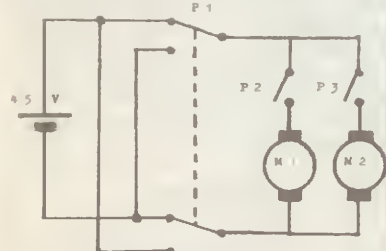
```
100 B=56577:P=100:SC=53280:POK
EB+2,255:POKEB,0:POKESC,0:POKES
C+1,0:POKE650,128
120 INPUT"DŁUGOSC  SEKWE
NCJI";D:IFD<1THEN120
140 IFD>50THENINPUT"NA PEWNO
(T/N)";A$:IFA$<>"T"THEN120
150 DIMV(D):INPUT"WPROWADZANI
E BINARNE CZY DZIESIETNE     (B
/D)";A$
155 IF(A$<>"B")AND(A$<>"D")THE
N150
165 PRINT"": FORI=1TOD:PRINTI,
A$;
180 IFA$="D"THENINPUTL$:V(I)=V
AL(L$)AND255
190 IFA$="B"THENINPUT"   000000
00",L$:GOSUB1000:V(I)
=BT
200 POKEB,V(I):NEXT:PRINT"
<+> SZYBCIEJ  <-> WOLNIEJ"
310 FORI=1TOD:POKEB,V(I):GETA$
:PRINTCHR$(19) A$:IFA$=CHR$(32)
THENRUN
340 IFA$="+"THENP=P-1:IFP<0THE
NP=0
350 IFA$="-"THENP=P+1:IFP>5000
THENP=5000
360 FORT=1TOP:NEXT:NEXT:GOTO31
0
1000 BT=0:FORK=0TO7:BI=VAL(MID
$(L$,8-K,1)):BT=BT+BI*2↑K:NEXT
RETURN

READY.
```

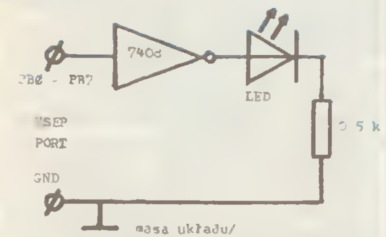
są nieco bardziej zaawansowanymi konstruktorami bądź programistami polecamy rozbudowanie układu, np. dodanie świateł, bezprzewodowego sterowania lub napisanie „mądrego" programu sterującego ruchem samochodziku.

Po zmontowaniu układu należy podłączyć samochodzik do komputera, włączyć zasilanie, wgrać program i... zacząć szaleństwo.

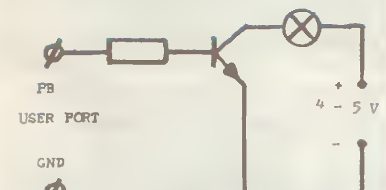
Ponieważ nie każdy ma możliwość skonstruowania jeżdżącego modelu samochodu, zamieszczamy jeszcze jedną propozycję — tu prace ograniczą się do wykonania banalnie wręcz prostego układu elektronicznego. Idea jest prosta — liczby dziesiętne, które wpisujemy do odpowiedniego rejestru w komputerze, pojawiają się na złączach USER PORT-u w postaci binarnej jako różne poziomy napięcia. Jeżeli do ośmiu szyn (PB0 — PB7) podłączymy diody świecące lub obwody żarówek, to wpisując do rejestru USER PORT-u kolejno liczby 1,2,4 ... 128 (binarnie 00000001, 00000010, 00000100 ... 10000000) uzyskamy efekt wędrującego światełka. Do podłączenia żarówek do komputera możemy wybrać schemat z rys. 3 lub 4. Rys.3 przedstawia podłączenie pojedynczych diod świecących, rys.4 — obwodów z żarówkami. Program obsługujący USER PORT pozwala wpisać dowolną ilość dziesiętnych lub binarnych wartości, odtwarzanych potem sekwencyjnie z wybraną szybkością.



Rys. 2 Połączenie przekaźników z silniczkami.



Rys.3 Schemat podłączenia diody świecącej do pojedynczej szyny USER PORTU. Zasilanie układu i diód z baterii 4.5 V. Szynę GND komputera i masę (−) układu należy połączyć. Układ 7408 można zastąpić układem 7400 ze zwartymi wejściami bramek.



Rys. 4 Schemat podłączenia układu z żarówkami do pojedynczej szyny USER PORTU. Przy podłączaniu ośmiu takich układów zasilanie może być wspólne

Wartości elementów:
R1 — 510 omów
R2 — 2.2 kiloomów
R3 — 100 omów
oporniki mogą być małej mocy (0.25 W)
T — tranzystory np. BC 107
US — układy 7400 lub 7408 (różnią się wprowadzeniem końcówek)
D — dowolna dioda na napięcie ok. 50 V
LED — dowolne diody świecące

Marek Bednarczyk
Michał Silski

Uwaga: używając powyższych programów z komputerem VC-20, należy odpowiednio zmienić adresy rejestrów sterujących USER PORTU.

PORADNIK MŁODEGO PIRATA — cz. V

Na zakończenie naszego „Poradnika" chcielibyśmy omówić pokrótce podstawowe sposoby zabezpieczania programów. Ze zrozumiałych względów nie będziemy się wdawać w szczegóły techniczne — chodzi nam raczej o zasygnalizowanie pewnych metod używanych do zabezpieczania.

Nie wierz nigdy drogi Czytelniku, że potrafisz zabezpieczyć program w sposób doskonały. Dla eksperta nie ma programu, którego nie dałoby się odbezpieczyć — jest to tylko kwestia czasu. Nie wierz również w sposobiki typu POKE 808, 234 i podobne, gdyż są one wartościowe jedynie w czasie działania tak zabezpieczonego programu; przed jego uruchomieniem (a żaden doświadczony programista nie będzie próbował „łamać" programu uruchomionego) można z nim zrobić co się chce. Istnieją sposoby na podglądanie zawartości pamięci, taśmy czy dyskietki; oprócz tego jest cała gama programów przeznaczonych przede wszystkim do „łamania" i (jeszcze większa) do kopiowania programów zabezpieczonych.

Najlepszymi znanymi nam sposobami zabezpieczania programów są:

1. Napisanie go w języku maszynowym. Konieczna jest do tego spora wiedza, ale jest też gwarancja, że byle programista nie wykreśli z Twojego programu imienia i nazwiska jego twórcy; będzie się musiał solidnie nabiedzić zanim uda mu się cokolwiek zrobić. Widzieliśmy już „piratów", którzy rezygnowali na sam widok instrukcji SYS.

2. Napisanie programu w BASIC i jego skompilowanie. Dzięki temu masz dwie widoczne korzyści: przyspieszenie jego wykonywania oraz zabezpieczenie przed włamywaczami. Takich kompilacji możesz dokonać w naszym klubie — zapraszamy! (adres klubu podany był w „Bajtku" nr 5-6 w kolumnie „SAMI O SOBIE"). Wystarczy nam do tego kaseta z nagranym programem, który chcesz skompilować. Przy okazji: dobrym zwyczajem jest przechowywanie gdzieś oryginalnej (nieskompilowanej) wersji programu.

3. Zaopatrzenie Twojego programu w procedurę samouruchamiającą. Bardziej oblatanym Czytelnikom podpowiadamy, że należy w tym celu zmienić wektor w komórkach 771 i 772 tak, aby wskazywał na początek Twojego programu. Innymi słowy, jeżeli napisałeś program (niestety dotyczy to prawie wyłącznie programów w kodzie maszynowym) zaczynający się dajmy na to od adresu 685 to wektor w adresach 771 i 772 musi „wskazywać" komórkę 685.

4. Zmiana wektorów systemu operacyjnego lub inne tego rodzaju pułapki. Bez odpowiednich zmian wektorów czy nawet procedur systemu program nie da się uruchomić.

Powyższe sposoby są bardzo szeroko stosowane, nie będą jednak one stanowiły większego problemu (no, może z wyjątkiem czwartego) dla specow od Commodore 64. Dla zupełnie „zielonych" i świeżo upieczonych programistów polecamy sposób drugi; jest to dla Was jedyna droga do zmiany BASIC na język maszynowy.

Poniżej chcielibyśmy przedstawić Czytelnikom kilka innych używanych sposobów zabezpieczania:

DYNAMICZNA KLAWIATURA

Powoduje wydruk na ekranie słowa RUN lub SYS + adres i niejako samoistne wciśnięcie sobie przez komputer klawisza RETURN; sposób ten używany jest jednak coraz rzadziej. Jeżeli gdzieś w loaderze trafisz na instrukcję POKE wpisującą jakieś wartości do komórek o adresach 631–640 to możesz być pewny, że jest to właśnie uruchamianie (właściwie — samouruchamianie) na zasadzie dynamicznej klawiatury. Istnieją dwie metody dezaktywacji — pierwsza z nich wprowadzenie do słowa RUN czy SYS błędu co spowoduje przerwanie wykonywania programu i sygnalizację błędu. Drugą metodą jest odszukanie instrukcji POKE 198, ... i wpisanie w zamian wartości 0. Bardzo często zmieniane są przy tym kolory tła i tekstu (na jednakowy, tak aby nie było widać co się dzieje na ekranie). Ponieważ metoda ta jest coraz rzadziej używana zdecydowaliśmy się opisać do niej sposób dezaktywacji; jako ciekawostkę chcielibyśmy podać, że w ten sposób była zabezpieczona jedna z wersji gry „BRUCE LEE".

ZBIORY

Zadaniem programu głównego jest wczytanie kilku lub kilkunastu zbiorów sekwencyjnych lub relatywnych z których składa się program główny; jest to jeden z wielu sposobów zabezpieczania na dyskach. W grupie tej warto wymienić procedury kontrolujące BAM czy określone sektory, zapis półsektorowy, wprowadzanie błędów na dysk, zapis niektórych procedur na ścieżkach o wyższym numerze niż 35, zmiany systemu operacyjnego stacji dysków i wiele innych. W niektórych programach (SKYFOX) w razie złej kopii wywoływana jest automatycznie procedura formatująca dysk; w innych operator jest proszony o włożenie oryginalnej dyskietki (PRINT SHOP, KENNEDY APPROACH).

„SUWAK"

Zabezpieczenie to polega na przesunięciu początku pamięci w adresach 43 i 44. Najczęściej po takim przesunięciu okazuje się, że program jest napisany w BASIC; zwykle w tak zabezpieczonym programie znajdują się jeszcze inne sposoby zabezpieczające.

NAKŁADKA

Generalnie rzecz biorąc jest to jak gdyby nałożenie na siebie dwóch czy trzech części programu w ten sam obszar pamięci. W pierwszej z nich znajduje się specjalna procedura maszynowa umożliwiająca takie nakładanie; gdzieniegdzie można spotkać jeszcze programy, w których program jest „rozstrzelony" np. co trzecią komórkę. Dwie pozostałe części wgrywane są w taki sposób, że po wgraniu trzeciej program jest gotowy do użytku.

STEROWANIE CZASOWE

Jest to jeden z lepszych systemów zabezpieczających — polega on na wczytywaniu programu w nieregularnych odstępach czasowych. W ściśle określonej chwili komputer oczekuje wczytania ściśle określonej ilości bajtów w ściśle określone miejsce pamięci. Odstępy czasowe są oczywiście dla utrudnienia bardzo nieregularne. Przykładem może tu być program BEACH HEAD w wersji kasetowej.

Oczywiście wspomniane wyżej systemy i procedury nie są jedynymi. Są one często zmieszane, dodatkowo wczytują się np. w obszar pamięci przeznaczonej dla stosu mikroprocesora, zmieniają wektory czy też nawet system operacyjny komputera. Czasami też są zabezpieczone za pomocą ... tytułu programu. Tak, tak to nie miraż — przecież po wczytaniu programu czy nawet tylko jego nagłówka wyświetlane jest tylko 16 pierwszych komórek (po komórkach zawierających rodzaj nagłówka, adresy wczytywania itp.), a ponadto gdy program się uruchamia dane te zostają w buforze. Wystarczy więc tylko umieścić w programie króciutką procedurę sprawdzającą zawartość bufora — i po sprawie ... Czasami zaraz za tytułem wczytywany jest krótki loader w języku maszynowym, który pozostaje w buforze; dla tych którzy opanowali dobrze zasady programowania w asemblerze i znają system operacyjny nie ma rzeczy nie do zrobienia.

Jeden z ciekawszych systemów opracowała firma ISEPIC; programy zabezpieczone przez tę firmę wczytywane są do RAM leżącej ... pod systemem operacyjnym. Jak wiadomo nic się długo w tej dziedzinie nie utrzyma — pojawił się w niedługi czas potem program ANTI ISEPIC umożliwiający szybkie odbezpieczenie programów tak zabezpieczonych.

Na zakończenie warto wspomnieć, że nauka programowania polega także na analizie programów napisanych przez innych. Celem jaki nam przyświecał podczas pisania takiego „pirackiego" poradnika była ochrona ludzi przed ich własną naiwnością — niejednokrotnie bowiem obserwowaliśmy jak są wykorzystywani świeżo upieczeni posiadacze komputerów przez np. niektóre wypożyczalnie czy handlarzy oprogramowaniem. Po co więc płacić, gdy część programów można przegrać samemu?

I jeszcze jedna uwaga. Nawet mały pirat nigdy nie podszywa się pod nazwisko twórcy programu. Jeżeli umie to się dopisze gdzieś w programie pozostawiając zawsze oryginalne nazwiska niezmienione; w tym cała jego „piracka" satysfakcja.

Klaudiusz Dybowski
Michał Silski

KLAN AMSTRAD-SCHNEIDER



AMSTRAD PC 1512

Niedawno w Wielkiej Brytanii przedstawiono nowy model komputera firmy AMSTRAD. Po serii przeznaczonej dla tych, ktorzy lubią gry, a przy okazji chcą mieć komputer do pracy i PCW (komputer dla sekretarki: duży monitor + drukarka + bardzo dobry program do edycji tekstów) AMSTRAD pokazał model PC1512 kompatybilny z IBM PC — absolutnym standardem na rynku mikrokomputerow profesjonalnych. W odróżnieniu od imitacji produkowanych przez państwa azjatyckie PC1512 nie jest tylko kolejną kopią. Widać charakterystyczne dla AMSTRAD-a podejście — dac użytkownikowi wszystko to, czego będzie potrzebował.

KLAWIATURA

Typowy układ 85 klawiszy — klawiatura alfanumeryczna numeryczna. 10 klawiszy funkcyjnych. Dodatkowo w obudowie znajduje się gniazdo do podłączenia joysticka

MONITORY

Tak jak w poprzednich modelach AMSTRAD-a monitor [illegible] dla całego komputera. [illegible] temu udało się zmniejszyć obudowę jednostki centralnej. Uzytkownik przy zakupie sprzętu ma do wyboru monitor monochromatyczny o wysokiej rozdzielczości lub kolorowy o rozdzielczości lepszej niz w serii CPC (tekst 80 znakowy jest czytelny), ale męczący przy dłuższej pracy.

AMSTRAD oferuje następujące typy graficzne. 16 kolorow i rozdzielczość 40 × 25 znaków (tylko tekst), 16 kolorów i 80 × 25 znakow (tekst), 3 palety po 4 kolory i 320 × 200 punktów (grafika), 2 kolory i 640 × 200 punktow (grafika) i wreszcie 16 kolorów przy rozdzielczosci 640 × 200 punktów (grafika)

DYSKI

Producent oferuje kilka wersji PC1512. W obudowie znajdują się dwa miejsca na stacje dysków — jedno zajmuje zazwyczaj stacja 5 1/4, a drugie mozna wykorzystać do wstawienia bądź drugiej stacji 5 1/4 bądź dysku typu WINCHESTER o pojemności 10 lub 20MB

JEDNOSTKA CENTRALNA

Komputer oparty jest o procesor Intel 8086 pracujący z zegarem 8MHz. Pamięć operacyjna ma pojemność 512kB z mozliwością rozszerzenia (na płycie głównej — nie zaj [illegible] cych systemem) do 640kB. Tu widac największe różnice w porównaniu z IBM PC. W IBM-ie pracuje procesor 8088 z zegarem 4,77 MHz. Mniejsza prędkość zegara oznacza, ze dla wykonania tej samej operacji IBM potrzebuje prawie dwa razy więcej czasu — jest znacznie wolniejszy. Poza tym 8088 jest okrojoną wersją 8086 — mikroprocesor jest 16-bitowy (rejestry wewnętrzne), lecz „z zewnątrz" wygląda tak, jak 8-bitowy — z otoczeniem moze się komunikować przez 8-bitową szynę danych i wszystkie operacje dotyczące pamięci wykonywane są na raty — najpierw jedna połówka słowa 16-bitowego, później druga. Jest to kolejne ograniczenie prędkości IBM PC w porównaniu z AMSTRAD-em. Średnio AMSTRAD jest 2-3 krotnie szybszy — zależy to od rodzaju wykonywanych operacji.

Na płycie głównej (motherboard) widać pozostałe różnice. To, co IBM traktuje jako dodatkowe rozszerzenia dostarczane za specjalną opłatą i zajmujące wolne sloty AMSTRAD uważa za standardowe wyposażenie komputera. RS232C, łącze szeregowe i CENTRONICS (zazwyczaj wykorzystywane do podłączania drukarki) wbudowane są na stałe. Również gniazdo do myszy i sama mysz „należy się kazdemu". Jedną z bardziej skomplikowanych spraw w komputerach IBM jest grafika i współpraca z ekranem. Uzytkownik może wybrac kartę tekstu, grafiki monochromatycznej, grafiki kolorowej i wiele innych. Kłopot z tym, że większość oprogramowania wymaga określonego typu karty — należy więc mieć wszystkie, co jest bardzo kosztowne. AMSTRAD oferuje rozwiązania kompromisowe, na płycie głównej znajduje się układ grafiki pozwalający symulować kartę tekstową, grafiki monochromatycznej i grafiki kolorowej. W ten sposób nie płacąc dodatkowo nic mamy prawie wszystko — brakuje jedynie karty grafiki kolorowej o bardzo wysokiej rozdzielczości, ale należy zdawać sobie sprawę, że sam monitor o wysokiej rozdzielczości kosztuje niewiele mniej niż cały AMSTRAD. Innym „niestandardowym" standardem jest zegar czasu rzeczywistego z zasilaniem bateryjnym — komputer nawet po wyłączeniu z sieci zna aktualny czas (przydatne przy programach typu kalendarz/sekretarka).

OPROGRAMOWANIE

Wraz z komputerem dostarczane są dwa systemy operacyjne MS-DOS 3.2 firmy MICROSOFT (większość IBM-ów używa starszej wersji 2.11) i DOS Plus firmy DIGITAL RESEARCH. DOS Plus podobny jest do znanego systemu Concurent CP/M86 pozwalającego na prace wieloprogramowe, prace w sieci i wykorzystanie większości programów systemów MS-DOS i CP/M86. Łącznie z systemem GEM (Graphic Enviroment Manager — Graficzne Zarządzanie Systemem) DOS Plus pozwala pracować w trybie takim jak na komputerze Mac Lntosh — wszystko przedstawione jest w postaci czytelnych piktogramów i praca z oprogramowaniem nie wymaga dokładnej znajomości maszyn. Na przykład pod kontrolą systemu MS-DOS chcąc wywołać program WordStar (edytor tekstów) piszemy „WS". W systemie GEM wskazujemy myszką obrazek maszyny do pisania — jest to o wiele prostsze i prz[illegible]e. GEM Paint (program graficzny) i GEM Desktop [illegible] wersji [illegible] kalkulator i kalendarz stanowią uzupełnienie systemu GEM. Uzytkownik otrzymuje również bardzo dobry interpreter BASIC-a (Locomotive) — o wiele szerszy i szybszy niż wersja MICROSOFT BASIC-a zawarta w EPROM-ie w komputerach IBM

NIEKOMPATYBILNOŚĆ Z IBM

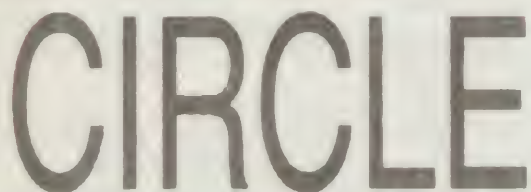
K[illegible] mogą wystąpić gdy program chce wykorzystać procedury z BASIC-a zapisanego w ROM-ie lub musi używać karty grafiki bardzo wysokiej rozdzielczości. Innego typu niespodzianki może sprawic większa prędkość działania AMSTRADA — wyobrazcie sobie dowolną grę zręcznościową przyspieszoną o 3 krotnie! — prawdopodobnie zanim się cokolwiek zacznie robic już jestesmy zabici i widzimy ulubiony napis GAME OVER. W zamian za to AMSTRAD wzbogaci rynek gier na komputery PC — już są szachy z bardzo dobrą trójwymiarową grafiką i świetna symulacja sportowa WINTER GAMES

Poza tym wyjątkami wszystkie programy z IBM PC działają bez zarzutow

PERSPEKTYWY NA PRZYSZŁOSC

Do tej pory AMSTRAD miał szczęście — jego komputery dobrzy przyjmowały się na rynku. Nowy model wygląda również dobrze, ale o tym co jest wart będzie się można przekonać dopiero po pewnym czasie. Na razie jest to jeden z najtańszych komputerów klasy IBM PC dodatkowo wyposażony w bogaty [illegible]

Dariusz Wichniewicz



Zamieszczony poniżej program służy do rozszerzenia BASIC-a Amstrada o dodatkową komendę CIRCLE.

Komendę tę wywołuje się przez pionową kreskę SHIFT + @ tak, jak np. instrukcje obsługi stacji dysków. Składnia instrukcji jest następująca:

CIRCLE,x,y,r

x,y — współrzędne środka okręgu

r — promień okręgu

Podobnie jak w Spectrum komenda ta rysuje okrąg o zadanych parametrach, nie zmienia jednak pozycji ostatniego narysowanego punktu. Współrzędne środka okręgu podlegają tym samym prawom, co współrzędne instrukcji PLOT i DRAW (tzn. ten sam zakres, ORIGIN).

Algorytm tej komendy w BASIC-u podajemy obok.

Loader w BASIC-u jest tak napisany, że zapewnia relokowalność programu tzn. można umieścić kod maszynowy pod różnymi adresami w pamięci. Adres początkowy należy umieścić w linii 260 przypisując go zmiennej a. Spośród linii 370–390 należy wybrać tę, która odpowiada posiadanemu modelowi Amstrada Schneidera. W liniach DATA ukryte są sumy kontrolne zapewniające kontrolę poprawnosci wpisywania programu. Jeżeli zdecydowaliśmy się na jeden adres początkowy, to po uruchomieniu loadera możemy podać komendę

SAVE "CIRCLE",b,a,495

i potem ładować komendę do pamięci rozkazami

MEMORY adr-1:LOAD "CIRCLE"

CALL adr

gdzie adr oznacza ustalony adres początkowy — dawna wartość zmiennej a.

Sergiusz Wolicki

```
10 REM Algorytm rysowania okregu
20 REM z komputera Spectrum
30 x=320:y=200:r=60
40 REM ustawienie wartosci poczatkowych
50 arcs=4*INT(INT(ABS(PI*SQR(r))+0.5)/4)
+4 ' ilosc odcinkow aproksymujacych
60 a=x+r:b-0:c=2*r*SIN(PI/arcs)
70 d=1-2*SIN(PI/arcs)^2
80 e=SIN(2*PI/arcs)
90 f=y-r*SIN(PI/arcs)
100 g=a
110 PLOT g,f
120 a=f
130 REM petla rysujaca
140 a=a+c:g=g+b
150 x=g:y=a
160 DRAW x,y
170 arcs=arcs-1:IF arcs=0 THEN STOP
180 h=b:b=b*d-c*e
190 c=h*e+c*d
200 GOTO 130
```

```
10 DATA 210900010D00C3D1BC000000001200C35F00434948
20 DATA 52434CC50000000000000000000000000000000A6
30 DATA 0000000000000000000000000000000000000000
40 DATA 000000000000000000000000000000000000B080
50 DATA 0000000081000000083A2DA0F4982FE03C021134F
60 DATA B17E3600F5E5CDC6BBE5D5CDCCBBE5D5DD4E02DD5F
70 DATA 460309EBDD4E04DD460509EBCDC9BBDD6E00DD6667
80 DATA 01CB7C28077C2F677D2F6F23AF111E00CD61BD21B1
90 DATA 23000605AF772310FCDD213501DD7E00B728074F47
100 DATA 060021140009DD23DD7E00B72B0A4F0600E52114F7
110 DATA 0009EBE1DD23DD7E00B72817DD23E521C8014F061A
120 DATA 00094E2346E1DDE5CD0E00DDE118C2D1E1CDC9BBD9
130 DATA D1E1CDC0BBE1F177C9D5E52AEC01EBE1CD16000793
140 DATA 30077D2F6F7C2F6723D1C9E5DDE1DD3404C9EB7300
150 DATA 2372C9EBC9FDE1DDE14E23460B79B02807702B71D4
160 DATA DD219501DDE5FDE5C9320A0832000D32460A32376F
170 DATA 1032000E32000F32410B32000E32000F32410A323F
180 DATA 41103200120005022346082332OB1E23081E0003D7
190 DATA 1E0009230009192308230A0A14230B230011140055
200 DATA 0319190A19000319 3C0C280A0828000100000052347
210 DATA 00010000072314102B0F102800010000052300001E8
220 DATA 000006320F082D19082D0F0A0F14080F1E0A0F2D81
230 DATA 0C14190A321E0A14321005000400000000000F900F5
240 DATA 16010F011D011B01F6BBC0BB5EBDA9BD82BD85BD8F
250 DATA 7FBD9ABD70BD64BD79BD8EBD67BD000000000000086
260 a=&A400:MEMORY a-1
270 FOR k=0 TO 24:READ a$:s=0
280 FOR l=0 TO 19:b=VAL("&"+MID$(a$,1+2*l,2)):POKE a
+20*k+l,b:s=s+b
290 NEXT l
300 IF RIGHT$(HEX$(s,4),2)<>RIGHT$(a$,2) THEN PRINT
"Blad w linii";(k+1)*10:END
310 NEXT k
320 DATA &1,&4,&d,&10,&9a,&a0,&ab,&b7,&c7,&d8,&fc,&1
2e,&1ca,&1cc,&1ce,&1d0,&1d2
330 FOR k=1 TO 17:READ b:b=a+b:c=PEEK(b)+256*PEEK(b+
1):c=a+c:POKE b,VAL("&"+RIGHT$(HEX$(c,4),2)):POKE b+
1,VAL("&"+LEFT$(HEX$(c,4),2)):NEXT k
340 READ b:POKE a+&63,VAL("&"+RIGHT$(HEX$(b,4),2)):P
OKE a+&64,VAL("&"+LEFT$(HEX$(b,4),2))
350 READ b:POKE a+&9D,b
360 FOR k=0 TO 10:READ b:POKE a+&1D8+2*k,b:NEXT k
370 DATA &b8f7,&40,&3d,&88,&61,&64,&5e,&79,&4f,&43,&
58,&6d,&46:'tylko dla CPC464
380 DATA &b113,&61,&5e,&a9,&82,&85,&7f,&9a,&70,&64,&
79,&8e,&67:'tylko dla CPC664
390 DATA &b113,&64,&61,&ac,&85,&88,&82,&9d,&73,&67,&
7c,&91,&6a:'tylko dla CPC6128
400 CALL a
```

RUCHOME LITERKI

Kilka dodatkowych linii na początku programu pozwoli na interesującą prezentację jego tytułu i autora. Wystarczy w tym celu, począwszy od linii 110, napisać odpowiedni tekst a$, który będzie wyświetlany w linii y ekranu. Podprogram w liniach od 30 do 100 realizuje ruch liter w następujący sposób:

Dla zmieniającej się wartości i tworzony jest z łancucha tekstowego a$ jednoznakowy łancuch b$=mid$(a$,i,1), wyswietlany następnie na pozycji j,y (linia 60) i kasowany spacją na poprzedniej jego pozycji j+1,y (linia 70). Linia 50 określa kolumnę j, w której ma być wyświetlony, znak począwszy od prawej strony ekranu. Ruch znaku w lewo zapewnia zamknięcie pętli w linii 80. Jednoczesnie w linii 50 człon 20-len (a$) /2+1 lokalizuje napis w środku ekranu. Linia 90 wprowadza małe opóźnienie w pojawianiu się kolejnych liter napisu. Możliwe jest również korzystanie z MOD 0 i MOD 2. Należy w tym celu zmodyfikować linię 20, wpisując zamiast MODE 1 inny wybrany tryb pracy. W linii 50 trzeba również uwzględnić ilośc kolumn ekranu zmieniając odpowiednio:

przy MOD 0 39 na 19 i 20 na 10
przy MOD 2 39 na 79 i 20 na 40

W. Ziółek

```
10 ' Animacja
20 CLS:MODE 1:GOTO 110
30 FOR i=1 TO LEN(a$)
40 b$=MID$(a$,i,1)
50 FOR j=39 TO 20-LEN(a$)/2+i STEP -1
60 LOCATE j,y:PRINT b$;
70 LOCATE j+1,y:PRINT " "
80 NEXT j
90 FOR d=1 TO 100:NEXT
100 NEXT i:RETURN
110 a$="--- Ruch ---":y=7:GOSUB 30
120 a$="--- liter ---":y=10:GOSUB 30
130 a$="--- na ---":y=13:GOSUB 30
140 a$="--- ekranie ---":y=16:GOSUB 30
```

KLAN ATARI

Jack Tramiel trafił po raz drugi. W branży komputerowej postęp jest tak szybki, że powtórzenie sukcesu rynkowego zdarza się niezmiernie rzadko. Przekonał się o tym choćby sir Clive Sinclair, a wiele opinii zamieszczanych w prasie fachowej wskazuje, że jego los może podzielić Commodore i jego najnowszy produkt. Tramiel był zresztą głównym architektem wielkiego sukcesu Commodore C-64. Zmienił jednak „barwy firmowe" na ATARI i jak do tej pory skutecznie strzela na bramkę, której kiedyś bronił. Najefektowniejszym a zapewne i najefektywniejszym z jego strzałów jest seria ATARI ST, która stała się prawdziwym przebojem rynku na lata 1985–1986.

Komputery serii ST zostały po raz pierwszy zaprezentowane w Las Vegas w styczniu 1985 r., na wielkich targach elektronicznych CES Consumer Electronics Show. Pokazano dwie maszyny: 520ST i 260ST. Różniły się one pojemnością pamięci RAM: odpowiednio 512 KB i 256 KB. Planowano także model 130ST, ale nie podjęto jego produkcji. Przyjęcie, z jakim spotkała się seria ST w Las Vegas i w kilka miesięcy później na targach w Hanowerze, potwierdziło, że hasło reklamowe Tramiela **„Power Without the Price"** w wolnym przekładzie — **„duży efekt małym kosztem"** nie jest tylko zręczną formułką, lecz znajduje potwierdzenie w rzeczywistości.

Prace nad serią ST podjęte zostały przez konstruktorów ATARI w maju 1984 r. Postawili oni przed sobą kilka celów. Pierwszym było znalezienie szybkiego mikroprocesora i pełne wykorzystanie jego możliwosci. Miało temu służyć rozszerzenie magistrali i uwolnienie mikroprocesora od jak największej ilości zadań, przez przekazanie ich innym układom. Po drugie, konstruktorzy nie zamierzali, jak powiedział jeden z szefów ATARI, ponownie wymyślać prochu, lecz raczej użyć najlepszych spośród wypróbowanych już rozwiązań. Trzecim celem było uzyskanie grafiki i kolorów najwyższej jakości. Wreszcie, zespół konstruktorski chciał dać ST możliwości rozbudowania systemu, poprzez wyposażenie go w szereg wejść i wyjść nie stosowanych dotychczas w komputerach domowych. Trzeba przyznać, że zamierzenia powyższe udało się ich twórcom zrealizować w stopniu przekraczającym wszelkie, a w każdym razie rynku i konkurencji, oczekiwania.



ATARI 520ST

MIKROPROCESOR

ATARI 520ST pracuje na mikroprocesorze **MC 68000** firmy **Motorola**. Układ ten jest pierwszym przedstawicielem nowej generacji procesorow 16-bitowych. Posiada 16-bitową szynę danych i 24-bitową szynę adresową. Częstotliwość zegara wynosi 8MHz.

Zapewnia ona komputerom ST bardzo dużą moc obliczeniową. Sporządzone przez Bruce'a Webstera, publicystę amerykańskiego miesięcznika BYTE, porównanie tej cechy komputerów: Macintosh, Macintosh Plus, Commodore Amiga i ATARI 520ST, wykazało zdecydowaną wyższość tego ostatniego.

PAMIĘĆ

ATARI 520ST ma **512 KB** pamięci RAM i 192 KB ROM. Wersja 260ST ma o połowę mniejszą pamięć RAM, natomiast 1040ST i 520ST+ mają po **1024 KB** RAM. Pamięć może być znacznie rozszerzona, gdyż mikroprocesor 68000 może adresowac bezpośrednio do 16 MB pamięci. Ogranicza tę wielkość układ kontrolujący pamięć, który może pracować „tylko" do 4 MB. Pamięć RAM składa się z 16 „kości" po 256 KB każda.

SYSTEM OPERACYJNY

System operacyjny ATARI ST, o nieprzypadkowej nazwie **TOS Tramiel Operating System**, umieszczony był początkowo w pamięci RAM. Nie dość, że zajmował w niej sporo miejsca, to w dodatku był uciążliwy dla użytkownika: odczytanie TOS z dyskietki trwało ponad pół minuty. Obecnie TOS znajduje się w ROM-ie. Oznacza to, że komputer jest gotów do pracy zaraz po włączeniu. W ROM-ie zawarty jest także GEM Graphics Environment Manager, system komunikacji graficznej z komputerem dzieło znanej firmy Digital Research. GEM wzorowany jest na doświadczeniach Macintosha, który zawdzięcza swoją popularność nowemu sposobowi łączności z użytkownikiem. GEM jest systemem typu WIMP Window, Icon, Mouse, Pushdown Menu — Okno, Ikon, Mysz, Rozwijane menu.

Pamięć ROM nie zawiera żadnego języka programowania. BASIC i LOGO znajdują się na dyskietkach, dołączanych wraz z podręcznikami do każdego komputera.

KLAWIATURA

ATARI ST ma klawiaturę składającą się z czterech części: klawiatura alfanumeryczna w układzie QWERTY, klawiatura numeryczna, 10 romboidalnych klawiszy funkcyjnych i klawisze sterujące ruchem kursora. Łącznie jest to **95 klawiszy**. Klawiatura jest bardzo wygodna w użyciu. Klawisz RETURN w kształcie odwróconego L jest trzykrotnie większy od pozostałych. Klawiaturą i gniazdami myszy/drążka sterowego zarządza mikroprocesor 6301, pracujący z częstotliwością 1MHz.

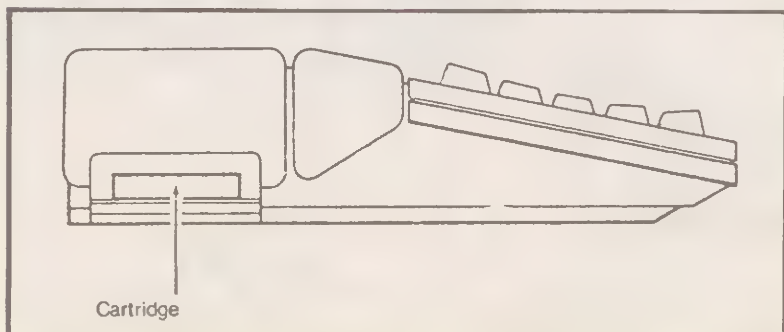
PERYFERIA

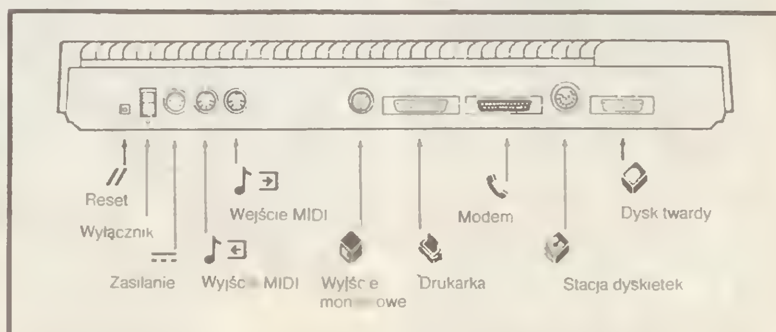
Z lewej strony komputera usytuowane jest gniazdo dodatkowych modułów pamięci ROM **cartridge slot**. Cartridge może mieć do 128 kB. Na tylnej ścianie umieszczono przycisk RESET oraz przełącznik zasilania. Znajduje się tam również duży zestaw łączy.

Łącze szeregowe RS-232C służy głównie do współpracy z modemem, a także z niektórymi typami drukarek. Łączność modemowa pomiędzy komputerami — za pośrednictwem linii telefonicznych — odbywać się może z prędkością od 50 do 19200 bodów bitów na sekundę. W praktyce stosowane są dwie prędkości: 300 i 1200 bodów.

Na szczególną uwagę zasługują dwa łącza szeregowe **MIDI Musical Instruments Digital Interface**. Nie są one z reguły stosowane w komputerach domowych. MIDI umożliwia współpracę z elektronicznymi instrumentami muzycznymi: zarówno z prostymi urządzeniami firm Casio czy Korg, jak i z profesjonalnym sprzętem Yamaha, perkusjami elektronicznymi itp. Istniejące oprogramowanie pozwala wykorzystać ST do celów profesjonalnych. Jednym z bardziej interesujących programów jest Pro 24, dzięki któremu komputer może spełniać rolę magnetofonu 24-ścieżkowego, kontrolować syntezator, służyć jako sequencer. Program ten, wraz z ATARI ST, wykorzystywany jest przez wielu muzyków, m.in. przez Marka Kelly, grającego na instrumentach klawiszowych w grupie Marillion.

Większość typów drukarek może być podłączana do łącza równoległego typu **Centronics**. Ogłoszenia polecają przede wszystkim firmową drukarkę SMM 804, a także





Panasonic, Star, Toshiba, Juki, NEC, Okidata, Epson, C. Itoh, Citizen, Legend.

Jedną z najciekawszych cech ATARI ST jest wbudowane gniazdo twardego dysku **hard disk** i układ bezpośredniego dostępu do pamięci **Direct Memory Access**. Używanie mikroprocesora do przenoszenia wielkich bloków danych z pamięci komputera do urządzeń peryferyjnych nie jest ani szybkie, ani wydajne. Układ DMA służy właśnie do realizacji takich szybkich transferów danych, bez angażowania mikroprocesora. W 1986 r. pojawiły się dwa rodzaje hard dysków do ATARI ST: 10 Mb i 20 Mb. Szybkość przekazywania danych wynosi 1.33 Mb/s. Godna uwagi jest cena hard dysku, która w przybliżeniu równa jest cenie zestawu ST ze stacją dyskietek i monitorem. Cena ta jest dwukrotnie niższa od ceny analogicznego, tej samej klasy, dysku sztywnego do Amigi.

STACJA DYSKIETEK

ATARI 520ST może współpracować z dwoma stacjami dyskietek standardu **3.5 cala**. Stacja SF354 ma zapis jednostronny, o podwójnej gęstości: na dyskietce mamy do dyspozycji 360 kB. Stacja SF314 ma zapis dwustronny: pojemność dyskietki jest dwukrotnie większa. Obsługa stacji dyskietek jest bardzo prosta: dyskietki można wyjąć po naciśnięciu guzika. Stacja dyskietek do ATARI ST jest znacznie mniejsza od, znanych choćby z „Pewexu", stacji ATARI 1050. Nowe 3.5-calowe dyskietki są znacznie wygodniejsze w użyciu, dzięki sztywnej obudowie z tworzywa sztucznego i jednocześnie o wiele trwalsze. Szybkość przekazywania danych wynosi 250 kbitów/s. Pracę stacji dyskietek kontroluje układ WD 1772 firmy Western Digital. Jest to zmieniona specjalnie dla ST wersja znanego już kontrolera WD 1770.

Istnieją wersje ST z wbudowaną stacją dyskietek. ATARI 1040ST z pamięcią RAM 1024 kB ma wbudowaną stację dyskietek i zasilacz.

ATARI 520ST może współpracować z **monitorem monochromatycznym SM124** lub **kolorowym SC 1224**. Monitor monochromatyczny zapewnia bardzo wysoką rozdzielczość, rzędu **640×400** punktów. Jasność i ostrość czarnobiałego obrazu sprawiają, że monitor ten jest niezwykle przydatny do celów profesjonalnych. Grafika monochromatyczna uzyskana dzięki ST reprezentuje rzeczywiście najwyższą jakość.

Drugi tryb graficzny, w którym można korzystać z 4 kolorów, daje rozdzielczość „średnia" medium 640×200 punktów.

Trzeci tryb graficzny, o „niskiej" low rozdzielczości, pozwala na uzyskanie rastra 320×200 punktów.

ATARI ST daje paletę — bagatela — **512 barw**. Powstają one w wyniku mieszania się ośmiu odcieni trzech kolorów: czerwonego, zielonego i niebieskiego Red, Green, Blue — RGB. Możliwości graficzne komputera są niezwykłe. Równać się z nimi może tylko Commodore Amiga. Każdy obraz na ekranie ST angażuje 32 kB pamięci RAM. Programy graficzne na ST, jak np D.E.G.A.S, czy NEOchrome doczekają się z pewnością osobnego omówienia w „Bajtku".

DŹWIĘK

Układ wytwarzający i kontrolujący dźwięk w ATARI ST nosi nazwę Yamaha YM2149. Ma trzy niezależne kanały dające dźwięk monofoniczny. Oprócz nich jest też generator szumów. Wewnętrzny zegar o częstotliwości 2MHz pozwala na uzyskanie tonów o wysokości od 30 Hz do 125 kHz.

ATARI ST sprzedawane jest w różnych konfiguracjach, w zależności od wersji. Wersje z modulatorem sprzedawane są bez monitora, a modele z wbudowaną stacją dyskietek bez tego urządzenia. Szkielet każdego zestawu składa się z komputera, myszy oraz dyskietek zawierających ST BASIC, ST LOGO wraz z podręcznikami, program graficzny NEOchrome i wordprocessor ist Word. Z dodatkiem stacji dyskietek i monitora monochrmatycznego lub kolorowego.

SUKCES

Sukces rynkowy serii ST jest bezprecedensowy. Nawet tak zwykle dotkliwie odczuwany przez nowe typy komputerów brak oprogramowania nie okazał się problemem dla ST. W każdym sprzedawanym zestawie zawarty jest program symulujący mikroprocesor Z80 działający pod nadzorem systemu operacyjnego CP/M. Otwiera on dostęp do ogromnej ilości programów napisanych dla systemu CP/M. W przygotowaniu znajduje się emulator IBM PC. Ma to być układ zawierający procesor Intel 8088 z 512 kB RAM, do którego będzie można dołączyć stacje dyskietek 5.25 cala. Układ ten będzie przyłączany do gniazda DMA.

Miarą sukcesu ST jest ogromne zainteresowanie firm piszących programy. Sam Tramiel, syn wielkiego Jacka i jeden z jego zastępców, powiedział, że oprogramowanie dla ST jest najszybciej rosnącą częścią rynku komputerowego i że zajmuje się tym ponad 1500 firm software'owych na całym świecie.

ATARI ST staje się na rynku komputerów domowych (nie zapominajmy, że to komputer domowy!) tym, czym przed kilku laty był C-64. Wiele przemawia za tym, że popularność ST będzie jeszcze większa. Sukces obydwu maszyn jest dziełem tego samego człowieka, Jacka Tramiela. Czy uda mu się powtórzyć go po raz trzeci?

Wiesław Migut
Jacek Barlik

DANE TECHNICZNE

MIKROPROCESOR: Motorola 68000, 16/32-bitowy, częstotliwość zegara 8MHz.
PAMIĘĆ: 512kB RAM
192 kB ROM
dodatkowo — 128 kB cartridge, 10 MB i 20 MB hard disk, CD-ROM
SYSTEM OPERACYJNY: TOS Tramiel Operating System
Sposób komunikacji z użytkownikiem: GEM Graphics Environment Manager
GRAFIKA: trzy tryby — 640×400 punktów, monochromatyczny, 640×200 punktów, 4 kolory; 320×200 punktów, 16 kolorów
KOLORY: 512
DŹWIĘK: trzy niezależne kanały, 30Hz do 125 Hz
KLAWIATURA: typ QWERTY, 95 klawiszy, klawiatura numeryczna, 10 klawiszy funkcyjnych, klawisze kontroli kursora.
STACJA DYSKIETEK: SF354 — dyskietki 3.5 cala, zapis jednostronny, 360kB, SF314 — dyskietki j.w., zapis dwustronny, 720 kB.
MONITOR: SM124 — monochromatyczny, SC 1224 — kolorowy RGB
ZŁĄCZA:

- **— monitora**
- **—Centronics dla drukarki**
- **— RS-232C dla modemu**
- **— stacji dyskietek**
- **— gniazdo hard disku DMA**
- **— gniazdo cartridge'a**
- **— gniazda myszy/joysticka**
- **— MIDI**

cz. II ŚWIAT DŹWIĘKÓW ATARI

Poznaliśmy już sposób generowania dźwięków za pomocą mikrokomputera. Spróbujemy teraz zająć się wytwarzaniem efektów dźwiękowych.

Gama efektow możliwych do zrealizowania przy użyciu Atari jest bardzo szeroka. Możemy naśladować dźwięki instrumentów muzycznych, odgłosy życia codziennego, samochodow, burzy, fal morskich, strzelaniny, wybuchow i fajerwerków. Wykorzystamy w tym celu instrukcje SOUND w języku BASIC. Uzyskane w ten sposob efekty są łatwe do modyfikowania. Dzwięk pozwoli uatrakcyjnic nasze programy i gry komputerowe. Warto mieć kilka gotowych programow z efektami dzwiękowymi. Atari może nam pomoc przy realizacji dzwięku do filmu amatorskiego.

Atari ma cztery niezależne generatory dźwięku o zakresie 3.5 oktawy każdy i dodatkowy generator szumów, który może być wykorzystywany przez generatory podstawowe jako zrodło zniekształcen. Jednoczesnie mozemy uruchomić maksymalnie cztery generatory. Istnieją możliwosci łączenia generatorow w pary w celu uzyskania większego zakresu, wykorzystywania filtrow itp. Niektore z nich zostały juz omówione. Obecnie korzystac będziemy tylko z podstawowych możliwości dźwiękowych Atari. Przypomnijmy, że w języku BASIC mamy do dyspozycji instrukcje SOUND. Posiada ona cztery parametry. Jej działanie trwa, dopóki nie zostanie podana następna instrukcja SOUND dla tego samego generatora, END, RUN lub NEW. Postac instrukcji SOUND jest następująca:

SOUND c,p,d,v

gdzie:

c — numer generatora od 0 do 3
p — okres generowanego dzwięku od 0 do 255
d — określa brzmienie liczby — parzyste od 0 do 14
v — jest głośnoscią od 0 do 15

Okres generowanego dzwięku okresla jego wysokosc. Zaleznośc pomiędzy wielkosciami określającymi częstotliwość jest następująca

częstotliwość=3196[illegible]res+1)Hz

gdzie liczba podająca [illegible]tliwosć jest z przedziału od 0 do 255 parametr p w instrukcji SOUND

Spróbujmy teraz zrealizować kilka efektow dzwiękowych. Mogą to być np. efekty pirotechniczne. Napiszemy i uruchomimy odpowiednie programy, omawiając za każdym razem role poszczególnych instrukcji. Rozpocznijmy od napisania i uruchomienia takiego programu

```
100 REM *** EKSPLOZJA ***
110 FOR A=1 TO 250
120 SOUND 0,A,2,6
140 NEXT A
150 SOUND 0,240,0,15
160 SOUND 1,245,0,15
170 SOUND 2,250,2,15
180 SOUND 3,255,4,15
190 FOR T=0 TO 50
200 NEXT T
210 FOR L=15 TO 0 STEP -1
240 SOUND 0,240,0,L
250 SOUND 1,245,0,L
260 SOUND 2,250,2,L
270 SOUND 3,255,4,L
280 FOR T=0 TO 30
290 NEXT T
300 NEXT L
310 END
```

Uzyskamy efekt dużej eksplozji. Program korzysta ze wszystkich czterech generatorów dźwięku. Pierwsza pętla zawarta w liniach 110-140 to „preludium" do wybuchu. Linie 150-180 realizują odgłos wybuchu. Cztery generatory pracują w zakresie najmniejszej częstotliwości, przy czym kazdy z nieco inną częstotliwością. Kazdy ma ponadto określone duze zniekształcenia. Linie 190-200 to pętla opóźniająca. Okresla ona czas trwania wybuchu. W liniach 210-290 zrealizowana jest pętla powodująca stopniowe wyciszanie odgłosow wybuchu.

Następny program realizuje odgłosy strzelaniny z broni maszynowej.

```
100 REM *** BRON MASZYNOWA ***
110 FOR A=1 TO 8
120 SOUND 0,175,6,14
130 C=INT(RND(0)*700+1)+50
140 GOSUB 200
150 SOUND 0,0,0,0
160 C=INT(RND(0)*200+1)+50
170 GOSUB 200
180 NEXT A
190 END
200 FOR T=1 TO C
210 NEXT T
220 RETURN
```

Pętla w liniach 110-180 realizuje osiem serii z broni maszynowej. Do generowania dźwięku wykorzystany jest tylko jeden generator linia 120. W linii 130 określany jest — przy użyciu generatora liczb losowych — czas trwania serii. Następnie wykonywany jest skok do podprogramu zawartego w liniach 200-220 który realizuje pętlę opóźniającą. Linia 150 wyłącza generator. W linii 160 określany jest czas trwania przerwy między seriami. Przerwa ta jest realizowana przez podprogram zawarty w liniach 200-220.

Odgłosy strzelaniny rewolwerowej da nam na przykład taki program:

```
100 REM *** STRZALY ***
110 FOR A=1 TO 8
120 FOR B=1 TO 70
130 SOUND 0,B,8,15
150 NEXT B
160 NEXT A
170 SOUND 0,0,0,0
180 END
```

Tak, jak poprzednio wykorzystuje on tylko jeden generator dźwięku. Pętla w liniach 110-160 realizuje osiem wystrzałów. Linie 120-150 to pętla zawierająca instrukcję sterującą generatorem, która zmienia częstotliwość generowanego dźwięku.

Spróbujmy teraz uzyskać odgłosy bombardowania

```
100 REM *** BOMBY ***
110 FOR A=1 TO 255
130 SOUND 0,A,14,10
140 NEXT A
150 FOR A=1 TO 350
160 SOUND 0,A,8,15
170 NEXT A
180 FOR A=1 TO 250
190 SOUND 0,A,8,15
200 NEXT A
210 SOUND 0,0,0,0
220 END
```

Znowu uzyskaliśmy efekty korzystając tylko z jednego generatora. Linie 110-140 to pętla, która realizuje odgłos spadających bomb. Zmienia ona częstotliwość generowanego dźwięku od najwyższej do najniższej możliwej do uzyskania za pomocą instrukcji SOUND. Następne dwie pętle w liniach 150-170 i 180-200 generują odgłosy wybuchającej serii bomb.

Jeszcze jeden przykład „pirotechnicznego" programu. Tym razem są to odgłosy fajerwerków.

```
100 REM *** FAJERWERKI ***
110 FOR A=1 TO 30
120 FOR B=1 TO INT(RND(0)*50+6)
130 SOUND 0,B,4,15
140 NEXT B
150 NEXT A
160 SOUND 0.0,0,0
170 END
```

[illegible] liniach 110 150 powoduje trzydzieści odgłosów [illegible] fajerwerków. Pętla 120-140 realizuje odgłos od[illegible]. Wartość maksymalna wskaźnika określana jest [illegible] liczb losowych. [illegible]

Eksperymentując z efektami akustycznymi pamiętajmy aby szanować nerwy [illegible]adów i nie słuchać ich przy pełnej mocy wyjściowej dźwięku z monitora. Nie powinny także być nadużywane programy realizujące sygnały karetki pogotowia, milicji czy straży pożarnej. Można jednak takie programy napisać. Zacznijmy od karetki pogotowia.

```
100 REM *** KARETKA ***
110 FOR A=1 TO 15
120 FOR B=30 TO 90
130 SOUND 0,B,14,10
140 GOSUB 230
150 NEXT B
160 FOR B=90 TO 30 STEP -1
170 SOUND 0,B,14,10
190 NEXT B
200 NEXT A
210 SOUND 0,0,0,0
220 END
230 FOR T=1 TO 2
240 NEXT T
250 RETURN
```

Program składa się z pętli w liniach 110-200, która powoduje piętnastokrotne powtórzenie sygnału. Pętle 120-150 i 160-190 realizują narastanie i opadanie częstotliwości generowanego dźwięku. Pierwsza z nich korzysta z podprogramu w liniach 230-250 który zawiera pętlę opóźniającą.

Dwa następne, bardzo podobne programy imitują sygnały radiowozu MO i straży pożarnej

```
100 REM *** RADIOWOZ MO ***
110 FOR A=1 TO 15
120 SOUND 0,50,14,10
130 SOUND 1,51,14,6
140 GOSUB 230
160 SOUND 0,73,14,10
170 SOUND 1,72,14,6
180 GOSUB 230
190 NEXT A
200 SOUND 0,0,0,0
210 SOUND 1,0,0,0
220 END
230 FOR T=1 TO 250
240 NEXT T
250 RETURN
```

```
100 REM *** STRAZ POZARNA ***
110 FOR A=1 TO 75
120 FOR B=1 TO 18
130 SOUND 0,50,10,8
140 SOUND 1,52,10,4
150 NEXT B
160 X=150
170 GOSUB 280
180 FOR B=1 TO 18
190 SOUND 0,70,10,8
200 SOUND 1,75,10,4
210 NEXT B
220 X=130
230 GOSUB 280
240 NEXT A
250 SOUND 0,0,0,0
260 SOUND 1,0,0,0
270 END
280 FOR T=1 TO X
290 NEXT T
300 RETURN
```

Podane przykłady dają wyobrażenie o możliwościach dźwiękowych Atari. Pole do eksperymentowania jest duże, a efekty — bardzo urozmaicone. W następnym odcinku „W świecie dźwięków Atari" postaram się przedstawić inne efekty dźwiękowe. Nie będą one aż tak „wystrzałowe", [illegible] ciekawe

Mariusz J. Giergiel



NOWE SPECTRUM

Na Dziewiątej Światowej Wystawie Komputerów Osobistych w londyńskiej Olimpii oznajmiono o opracowaniu nowego komputera firmy SINCLAIR SPECTRUM 128K + 2. Jest to pierwszy komputer wyprodukowany przez f-mę SINCLAIR'a po wykupieniu jej przez AMSTRAD-a.

Nowy Spectrum posiada klasyczny dla Amstrada układ — wmontowany magnetofon. Z boku komputera wbudowano również dwa porty do podłączenia drążków sterowych oraz wyłącznik do kasowania pamięci (RESET). A teraz kilka informacji szczegółowych. Pojemność pamięci RAM 128 kilobajtów, pamięć ROM — 32 kilobajty. Mikroprocesor firmy Zilog Z80A. Grafika 256 na 192 punkty na ekranie ... 16 kolorów do dyspozycji użytkownika. (Osiem zasadniczych i osiem w tle). Trzykanałowy układ dźwiękowy i znane z AMSTRADA obwiednie. System kontroli jakości programów, opracowany przez firmę SINCLAIR QUALITY CONTROL (SQC). Wreszcie komputery te mają w pełni profesjonalną klawiaturę typu QWERTY o 58 klawiszach, wśród których tylko nieliczne zachowały stare funkcje i są opisane pełnymi komendami jak RUN, EDIT czy LOAD.

Jak podaje wytwórca, od 1982 roku sprzedano ponad dwa miliony komputerów firmy SINCLAIR. Były wtedy prawdziwą rewelacją. Trudno jednak wycofać je z rynku właśnie ze względu na ilość sprzedanych egzemplarzy i olbrzymią ilość oprogramowania.

Jak stwierdzono w informacji dla prasy, komputer ten budowano nie z myślą o miłośnikach gier, a raczej o poważniejszych zastosowaniach.

Można do niego przyłączyć seryjną drukarkę, kolorowy monitor RGB, znaną z modelu 128 klawiaturę numeryczną, modem i wiele innych urządzeń peryferyjnych, których wymagają dzisiejsze komputery domowe. Dla użytkowników o wyrobionym smaku muzycznym przewidziano możliwość podłączenia syntetyzera muzycznego (poprzez port MIDI) i odtwarzania (przez wzmacniacz stereofoniczny) skomponowanych symfonii.

SPECTRUM 128K+2 jest kompatybilny z INTERFACE 1. Wspomniany wyżej system kontroli jakości SQC oznacza, że programy opatrzone tym skrótem zostały sprawdzone przez fachowców [illegible] cią będą działały w nowym komputerze.

Nim nowe dziecko Sinclair'a pokaże się na polskim rynku [illegible] informacją

Tomasz Pyć

STRAŻNIK BANKU

Po długiej podroży dotarłeś do Gun City. Celem wędrowki było zdobycie pracy i szczęśliwym trafem tutejszy bank pilnie potrzebował strażnika. Otrzymałeś służbowego colta i usiadłeś sobie w kącie pilnie bacząc na to kto wchodzi. Drzwi się otwierają i ... wchodzi jakaś dama pragnąc wpłacić swoje oszczędności.

Przez drugie drzwi znienacka usiłuje wtargnąc jakis oprych. Wyciąga pistolet ale ty jesteś szybszy. Nagle w drugich drzwiach ukazuje się żądny krwi bladych twarzy Indianin, ale już po chwili jego duch wędruje do Manitou. Znowu byłeś lepszy. I znowu ktoś otwiera drzwi. Strzelasz bez zastanowienia i Boo! ducha winny farmer wita sie ze Sw. Piotrem. Lecz na szczęście tutejszy szeryf jest pobłażliwy i zawsze mowi: „do trzech razy sztuka". Ktoś wchodzi. Teraz uważnie go oglądasz. Błąd! Jego słow — „Jestem szybszy!" słuchasz już w towarzystwie swiętego Piotra.

Tak, szanowny czytelniku, może wyglądać i twoja historia jeśli zadasz sobie trud i mozolnie „wstukasz" poniższy program do Twojego Spectrum. Teraz dam Ci kilka praktycznych wskazowek. Duże litery w cudzysłowach w liniach 70—410 należy wprowadzac w trybie graficznym, tzn. C.SHIFT + 9. Wówczas na ekranie otrzymamy kursor "G". Powrot do normalnego kursora odbywa się analogicznie. Mozna zmienic wartosc punktacji przez zamianę w linii 4020 wartosci 100 na taką, jaka Ci będzie odpowiadała. Możesz efekt strzału zaczerpnąc z 1--go numeru nowego Bajtka. Możesz robić z tym programem co tylko Ci się podoba, bowiem żadne prawa autorskie nie są zastrzeżone. Acha, jeszcze dla ułatwienia dodam, że im wyższą liczbę podasz przy pytaniu o stopień utrudnienia, tym łatwiej będzie Ci się grało. Radził bym jednak nie przesadzac, gdyż przy b. wysokiej liczbie program staje się powolny i po prostu nudny.

Sterowanie odbywa się klawiszami 1,2,3; klawisz 1 odpowiada pierwszym drzwiom od lewej itd.

Dobrego refleksu!

Grzegorz Zatryb

```
   1 BORDER 0: INK 7: PAPER 0: CLS
   5 REM Wczytywanie nowych znakow UDG
  10 FOR f=0 TO 167
  20 READ a
  30 POKE USR "a"+f,a
  40 NEXT f
  50 DATA 0,0,19,11,3,3,1,1,0,192,200,248,192,192,192,128,15,11,11,11,11,11,11,11
  51 DATA 240,208,208,208,208,208,208,208,3,6,12,24,48,32,32,224,192,96,48,24,12,4,4
  52 DATA 15,107,43,121,35,115,115,115,15,11,11,11,235,69,3,224,16,208,208,208,144,128,248
  53 DATA 236,196,140,152,176,128,192,192,192,224,240,248,252,254,64,96,7,6,11,11,11,9,1,31
  54 DATA 35,35,33,61,1,1,3,3,7,15,31,63,127,2,6,64,72,224,208,208,192,128
  55 DATA 255,225,225,225,129,216,224,96,48,24,24,24,24,24,8,15,17,5,19,15,19,1,1
  56 DATA 221,215,87,87,87,119,71,6,12,24,24,24,24,24,16,240,255,129,189,189,189,189,129,255
  57 REM Koniec wczytywania
  58 GO TO 8000
  59 GO TO 430
  60 REM 1 czlowiek
  65 LET kontr=0
  70 PRINT AT x,y;"AB"
  80 PRINT AT x+1,y;"CD"
  90 PRINT AT x+2,y;"EF"
 100 RETURN
 110 REM kobieta
 115 LET kontr=0
 120 PRINT AT x,y;"LI"
 130 PRINT AT x+1,y;"MJ"
 140 PRINT AT x+2,y;"NK"
 150 RETURN
 160 REM indianin
 165 LET kontr=1
 170 PRINT AT x,y;"RO"
 180 PRINT AT x+1,y;"SP"
 190 PRINT AT x+2,y;"TU"
 200 RETURN
 210 REM 3 czlowiek
 215 LET kontr=0
 220 PRINT AT x,y;"AB"
 230 PRINT AT x+1,y;"GD"
 240 PRINT AT x+2,y;"EF"
 250 RETURN
 260 REM 4 czlowiek
 265 LET kontr=1
 270 PRINT AT x,y;"AB"
 280 PRINT AT x+1,y;"HC"
 290 PRINT AT x+2,y;"EF"
 300 RETURN
 310 REM Zamykanie drzwi
 320 PRINT AT a,b;"  "
 330 PRINT AT a+1,b;"  "
 340 PRINT AT a+2,b;"  "
 350 PRINT AT a+3,b;"  "
 360 RETURN
 370 REM Otwieranie drzwi
 380 PRINT AT a,b;"  "
 390 PRINT AT a+1,b;"  "
 400 PRINT AT a+2,b;"  "
 410 PRINT AT a+3,b;"  "
 420 RETURN
 430 REM Rysowanie tla
 435 CLS
 440 PLOT 0,56
 450 DRAW 8,16
 460 DRAW 239,0
 470 DRAW 8,-16
 480 DRAW 0,-42
 490 DRAW -8,16
 500 DRAW -239,0
 510 DRAW -8,-16
 520 DRAW 0,40
 530 PLOT 8,32: DRAW 8,-32: DRAW 223,0: DRAW 8,32
 540 PLOT 88,32: DRAW -24,0,PI
 550 PLOT 144,32: DRAW -24,0,PI
 560 PLOT 200,32: DRAW -24,0,PI
 570 PLOT 48,32: DRAW 0,40: PLOT 144,32: DRAW 0,40
 580 PLOT 160,32: DRAW 0,40: PLOT 216,32: DRAW 0,40
 590 PLOT 8,32: DRAW 0,40: PLOT 247,32: DRAW 0,40
 600 REM Zmiana atrybutow w pierwszych trzech liniach
 610 FOR i=22528 TO 22623
 620 POKE i,22
 630 NEXT i
 640 REM Ramka
 650 PLOT 0,175: DRAW 255,0: DRAW 0,-24: DRAW -255,0: DRAW 0,24
 670 PLOT 4,171: DRAW 96,0: DRAW 0,-16: DRAW -96,0: DRAW 0,16
 680 PLOT 196,171: DRAW 32,0: DRAW 0,-16: DRAW -32,0: DRAW 0,16
 700 PRINT INK 6; PAPER 2;AT 1,1;"PUNKTY"
 710 PRINT INK 6; PAPER 2;AT 1,25;"AB"
 720 INK 5: PLOT 88,136: DRAW -25,0: DRAW 0,-33: DRAW 25,0
 730 PLOT 144,136: DRAW -25,0: DRAW 0,-33: DRAW 25,0
 740 PLOT 200,136: DRAW -25,0: DRAW 0,-33: DRAW 25,0
 745 REM Rysowanie drzwi
 750 INK 7: LET a=5: FOR b=8 TO 22 STEP 7: GO SUB 320: NEXT b
 760 REM Dane
 770 LET pkt=0
 780 LET a=5
 790 LET spr=4: LET b=8
 800 LET oh=4
 810 LET zyc=3
 820 LET licz=0
 890 REM Glowny program
 900 LET i$=INKEY$
 905 LET oh=4
 940 LET los=INT (RND*5)
 950 LET q=60+los*50
 960 LET dl=INT (RND*3)
 961 IF spr=dl THEN GO TO 965
 962 GO SUB 320
 965 LET spr=dl
 970 PRINT AT 1,8;pkt: PRINT AT 1,27;zyc
 975 LET b=8+dl*7
 980 LET y=8+dl*7
 990 GO SUB 370
 995 LET licz=0
1000 GO SUB q
1005 GO SUB 6000
1006 LET i$=INKEY$
1010 IF i$="1" THEN LET oh=1: GO SUB 3000
1020 IF i$="2" THEN LET oh=2: GO SUB 3000
1030 IF i$="3" THEN LET oh=3: GO SUB 3000
1035 IF licz=utr AND kontr=1 THEN GO TO 7000
1036 IF licz=utr THEN GO TO 900
1050 GO TO 1005
1060 GO TO 900
1999 REM Trafiony niewinny.
2000 PRINT AT a+2,b;"*"
2010 BEEP .3,-10
2020 LET zyc=zyc-1
2030 IF zyc=0 THEN GO TO 7000
2035 LET oh=4: LET kontr=2
2036 LET licz=0
2040 RETURN
2999 REM Kto trafiony ?
3000 IF oh=dl+1 AND kontr=0 THEN GO SUB 2000
3010 IF oh=dl+1 AND kontr=1 THEN GO SUB 4000
3020 RETURN
3999 REM Trafiony bandyta.
4000 PRINT AT a+1,b;"*"
4010 BEEP .3,-10
4020 LET pkt=pkt+100
4025 LET oh=4: LET kontr=2
4026 LET licz=0
4030 RETURN
5999 REM Licznik czasu.
6000 LET licz=licz+1
6010 RETURN
7000 REM koniec gry
7010 PRINT FLASH 1;AT a-1,b+1;"JESTEM";AT a,b+1;"SZYBSZY"
7035 PAUSE 150
7040 CLS
7050 PRINT AT 11,1;"Masz ";pkt;" pkt."
7060 PRINT AT 13,1;"Jeszcze raz (t/n) ?"
7070 IF INKEY$="t" THEN GO TO 59
7080 IF INKEY$="n" THEN RANDOMIZE USR 0
7090 GO TO 7070
7999 REM Poczatek gry
8000 INPUT "utrudnienie ";utr
8010 IF utr<0 AND utr>100 THEN GO TO 8030
8020 GO TO 8000
8030 PRINT AT 11,10;"GOTOW"
8040 PAUSE 0
8050 GO TO 59: REM skok do poczatku glownej petli.
```

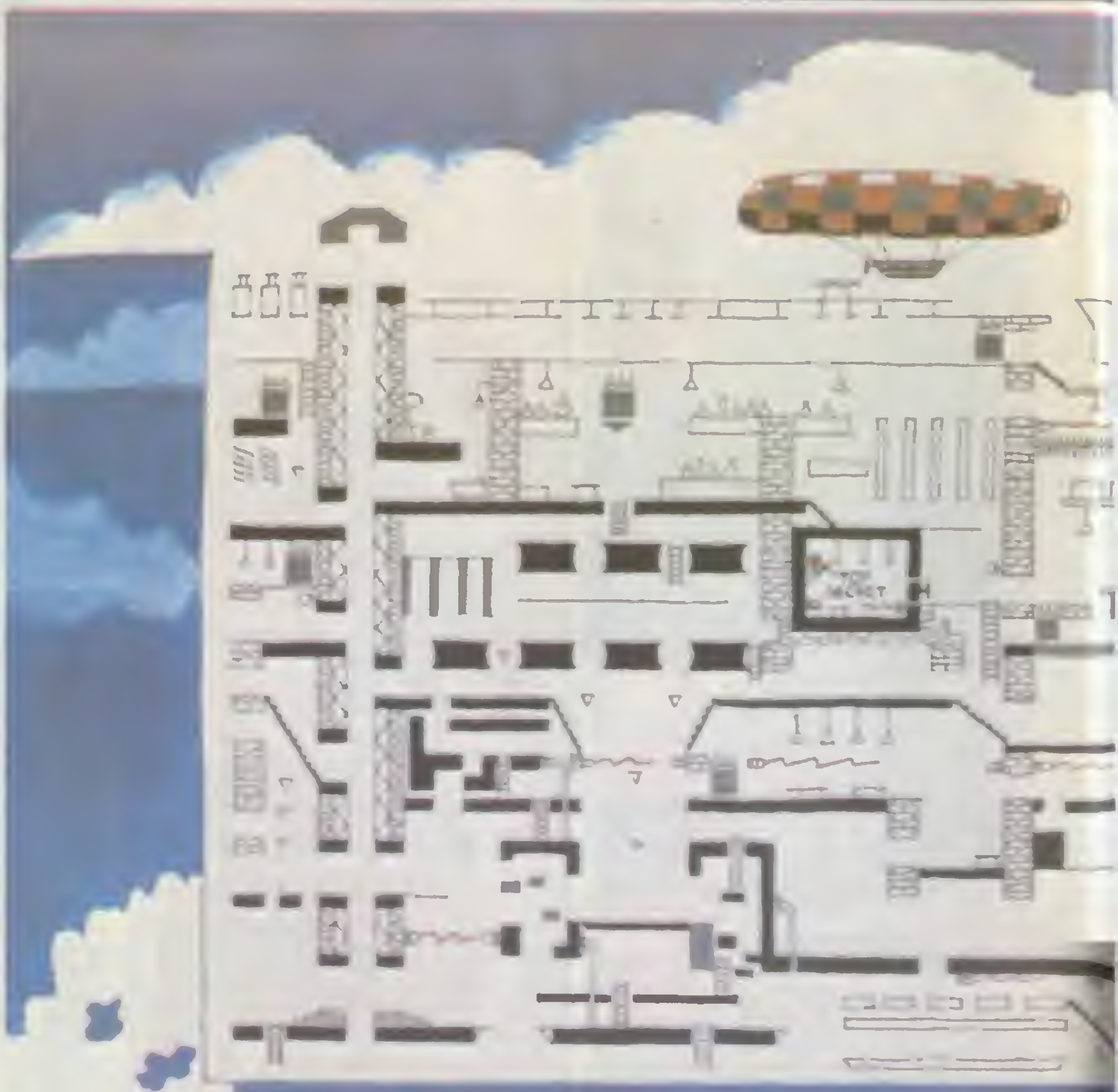
ŻYCZENIA OD KOMPUTERA

Popatrzcie na ten malenki program. Wiecie już jaki będzie efekt jego działania? Jeśli nie, to koniecznie go wpiszcie i przekonajcie się sami. Cała tajemnica jest oczywiście ukryta w linii 20. Spróbujcie odgadnąć co ukrywa się w komórce pamięci o numerze 23606. Jeśli się to wam nie uda, zaglądnijcie do podręcznikow obsługi Spectrum lub też do poprzednich numerow „Bajtka".

(rp)

```
10 FOR i=255 TO 0 STEP -1
20 POKE 23606,i
30 PRINT AT 10,5; "Moc zabawek spod choinki"
40 PRINT AT 12,5; "Dla Kubusia i Balbinki."
50 PRINT AT 14,18; "Bajtek"
60 NEXT i
```

DYNAMITE DAN



Przybywasz do ogromnego laboratorium. Twoim zadaniem jest wykraść tajne dokumenty profesora, znajdujące się w kasie pancernej. Gdy już szczęśliwie zdobędziesz papiery z napisem TOP SECRET musisz wrócić do swojego sterowca, na którym opuścisz to niebezpieczne miejsce.

Aby otworzyć kasę potrzebujesz 8 lasek dynamitu, które znajdziesz wędrując po laboratorium. Do Twojej dyspozycji pozostawiono windę, teleporty i tratwę. Teleporty są sprzężone ze sobą parami. Każdy z nich przenosi Cię w sprzężony z nim teleport bliźniaczy. Tratwa pływa kanałem pod laboratorium, zeskakujesz na nią z jednego z najwyższych poziomow dachu.

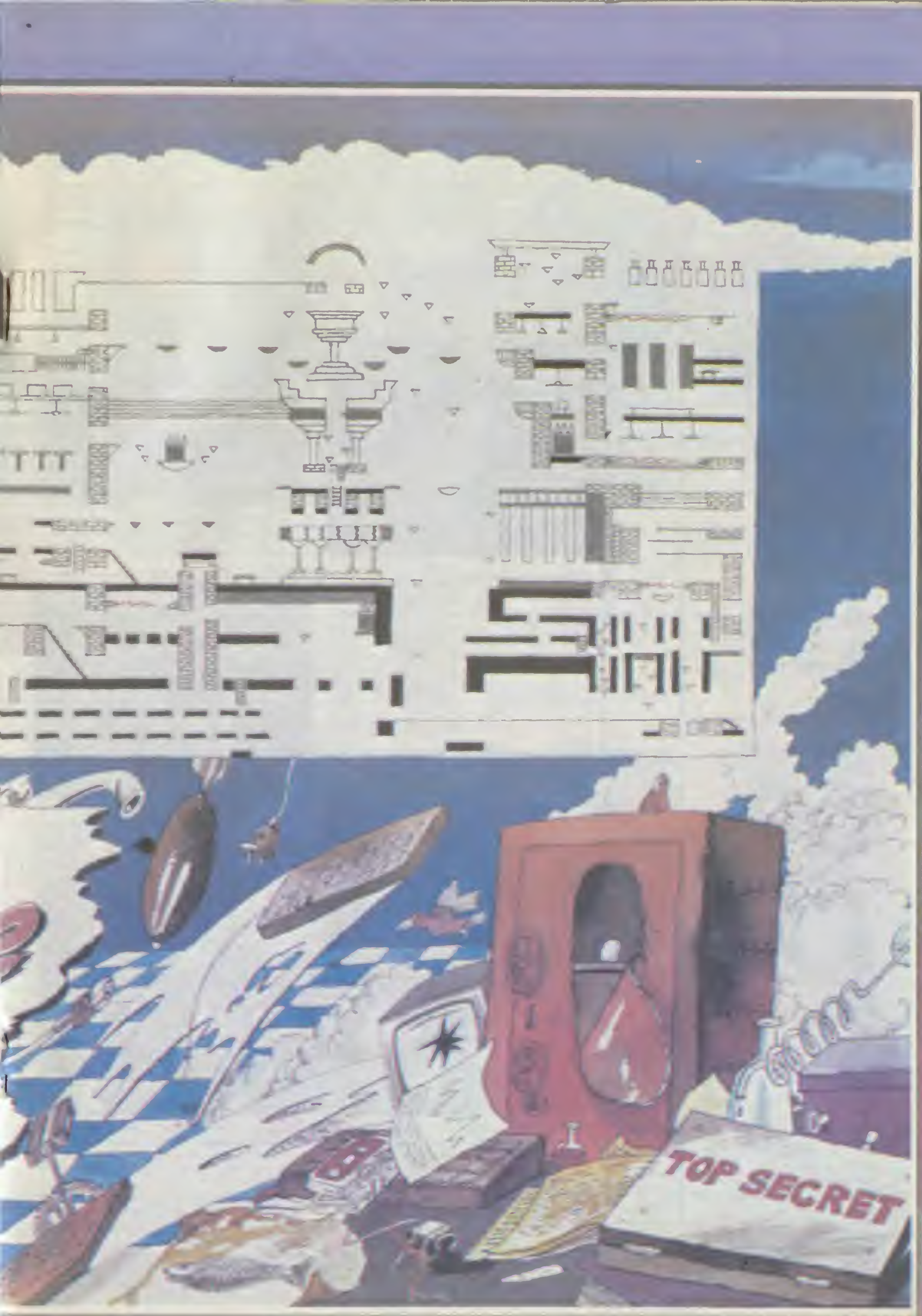
Niebezpieczenstwa czyhają na Ciebie wszędzie, także podczas płynięcia tratwą, łatwo stracic głowę i spaśc z wąskiej deski, wtedy uratować Cię przed zatonięciem (co jest równoznaczne z zakończeniem gry) może zasobnik z tlenem, ktory znajdziesz w doskonale zaopatrzonym laboratorium.

Wędrówka jest męcząca, możesz jednak regenerowac siły i energię spożywając smakołyki przygotowane w różnych miejscach, jednakże musisz strzec się niebezpieczenstw, których jest znacznie więcej, przybierają one rozmaite kształty i mają wspólną cechę — pozostają w ciągłym ruchu. Rozmaite potworki nie są już tak groźne, gdy znajdziesz Deodorant, ktory pozwoli Ci je zniszczyc bez utraty energii. Pomoc Ci także może, błyszcząca życiodajnym blaskiem probówka. Poślizgnięcie się i upadek do wody może zakonczyc Twoją grę.

Jeśli bawił Cię JET SET WILLY, na pewno spoboba Ci się DYNAMITE DAN — jest to gra bardziej atrakcyjna i nie tak nużąca.

Dynamite Dan...

Bajtek


TOP SECRET

10 BAJTKOWA LISTA PRZEBOJÓW (9)

Gwiazdkowe notowanie Bajtkowej Listy Przebojów cieszyło się największą jak dotychczas popularnością. Głosowało 3218 osób na ponad 500 tytułów. A tak wygląda notowanie 9.

			AMSTRAD	ATARI	COMMODORE	SPECTRUM
1	SPY Vs SPY	▲				
2	ELITE	▲				
3	ROAD RACE	▲				
4	ZORRO	▲				
5	CAULDRON	!				
6	BOMB JACK	▼				
7	BACK TO SKOOL	▼				
8	LAST V8	!				
9	BOULDER DASH	!				
10	MOVIE	▼				

W tabelce przedstawiamy na jakie komputery są dostępne w Polsce programy prezentowane na liście.

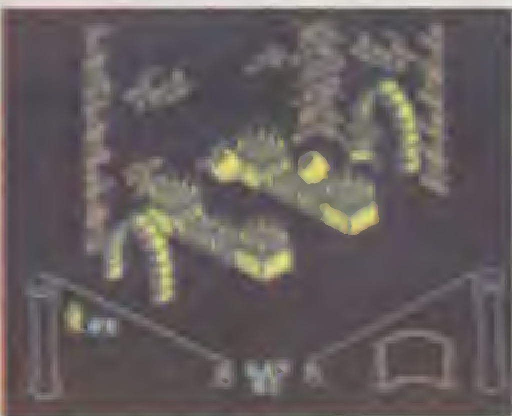
Tym razem przyznajemy nagrody za najlepsze opisy typowanych gier. Nagrody — kasety z programami [illegible] ufundowane przez brytyjską firmę ELECTRONICS EXPORT otrzymują: Tomasz Fo[illegible] z Warszawy, Andrzej D[illegible]iec z Krakowa i W[illegible] Peryśko z Gdańska.

Sławek

ZŁOTA DZIESIĄTKA ROKU 1986

Dziś w naszym świątecznym numerze przedstawiamy podsumowanie za rok 1986 notowań Bajtkowej Listy Przebojów.

Za pierwsze miejsce na każdej liście przyznaliśmy 10 punktów, za ostatnie — 1 punkt. Zsumowanie wszystkich punktów dało nam Złotą Dziesiątkę.



1 KNIGHT LORE — **ULTIMATE** (Spectrum, Amstrad) — ***to już historia.***

Pierwsza z długiej „serii" gier trójwymiarowych, graficznie nienaganna, gra która przeszła do „historii" dzięki pomysłowemu pozbyciu się atrybutów (na ZX Spectrum). Labirynt który mały Rycerz musi penetrować w poszukiwaniu potrzebnych Wielkiemu Magowi przedmiotów jest dość rozległy i czyha w nim dużo niebezpieczeństw. Gdy przyjdzie noc mały Rycerz zmienia się w wilkołaka, to czar rzucony przed laty nie daje mu spokojnie spać. Gra może dać wiele radości, szczególnie dzieciom.



2 BOMB JACK — **ELITE** (Spectrum, Commodore) — ***trochę za mało znany, a szkoda.***

Choć też Jack to już nie tak słaby jak Jumping. Gra bardzo dynamiczna, w której refleks znaczy bardzo dużo. Grafika chyba najlepsza z gier typu arcady. Prosta ale wymagająca od gracza pewnego sprytu dla przechytrzenia przeciwnika.

3 THE WAY OF THE EXPLODING FIST — **MELBOURNE HOUSE** (Spectrum, Commodore, Amstrad) — ***pierwszy udany.***

Dla zwolenników walk wschodnich sparring z gwardią Wielkiego Mistrza na zasadzie walk sportowych. Niezła grafika i płynność ruchów obu zawodników, sprawiają, że gra wciąga i można w nią grać godzinami. Dzięki [illegible] c[illegible]ów m[illegible] swoją taktykę [illegible]

4 JUMPING JACK — **IMAGINE SOFTWARE** (Spectrum) — ***pełna niespodzianka.***

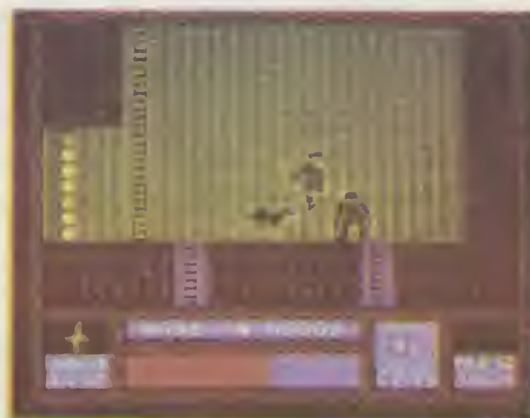
Jedna ze słabszych gier pod każdym względem. Monotonna, uboga graficznie, posiada nieskomplikowane efekty dźwiękowe. A jednak czytelnicy dwukrotnie umieścili ją na pierwszym miejscu.

5 ROAD RACE — **ACTIVISION** (Atari, Commodore) — ***gaz do dechy.***

Gra łącząca elementy symulacji samochodu wyścigowego z wielką przygodą. Samochodowy rajd przez całe Stany Zjednoczone od zachodniego do wschodniego wybrzeża — to trzeba po prostu przeżyć.

6 IMPOSSIBLE MISSION — **US GOLD-EPYX** (Spectrum, Commodore) — ***ma lepszą dykcję niż spiker w TV.***

Trudna do konkretnego sklasyfikowania gra w połowie zręcznościowa, w połowie układanka Puzzle. Wymaga zdolności abstrakcyjnego myślenia i przewidywania, gdy brak graczowi tych atrybutów gra może okazać się trudna. Grafika w normie, ale dźwięk, szczególnie z Commodore, jest rewelacyjny — bardzo udana synteza mowy.



7 SABOTEUR — **DUREL SOFTWARE** (Spectrum, Amstrad, Commodore) — ***taki program i tak nisko.***

Jak każdy wytwór firmy Durel Software jest warty 100 funtów za wskazanie kogoś kto kopiuje ten program, ale... W określonym czasie trzeba dojść do centrum komputerowego, podłożyć bombę, zabrać dysk z pamięci komputera i uciec śmigłowcem. Proste na pierwszym poziomie ale co zrobić, gdy drzwi są pozamykane? Świetna grafika i animacja.

8 HOBBIT — **MELBOURNE HOUSE** (Spectrum, Commodore, Amstrad) — ***najpierw przeczytaj potem zagraj.***

Któż nie zna bohaterów książek J. R. R. Tolkiena. Przygody Hobbita są zgodne z książką, ale przejście je na ekranie to wielka sztuka. Hobbit to już klasyka, ale zawsze ma zwolenników.

9 FAIRLIGHT — **THE EDGE** (Spectrum, Amstrad) — ***czy można to wygrać?***

Najlepsza chyba do tej pory muzyka na ZX Spectrum. Grafika też warta jest zobaczenia. Jedyną wadą jest zbyt długi czas generowania obrazu. Gra dość trudna, trzeba wielu pomysłów aby odszukać niektóre przedmioty.

10 TIR NA NOG — **GORGOYLE GAMES** (Spectrum, Commodore) — ***zagadka nie do rozwikłania.***

Jak dotąd nigdzie nie było kompletnego planu do tej gry z zaznaczeniem wszystkich tajnych przejść. Zmusza do myślenia. Ciekawa animacja i nienaganna grafika stawia ten program na pewno wyżej niż [illegible] [illegible] przy której [illegible] się nudzić.

OSTATNI V8



Smutna to gra (choc zapewne lepiej w nią „grać" na niby, niż naprawdę), ale za to doskonała dla tych, ktorzy uwielbiają emocje najwyższego stopnia.

„V8, V8 wracaj natychmiast do bazy" — wzywa głos w słuchawce i oto zaczyna się Twoja przygoda. Musisz zrobić wszystko, żeby zdążyć do bazy zanim..., ale zacznijmy od początku.

Jako wybitny naukowiec, pracujący nad projektem V8, w czasie, gdy nastąpiła nuklearna zagłada, ukryty byłeś w podziemnej bazie. Teraz otrzymałeś pozwolenie, aby wyjść na zewnątrz i sprobowac nawiązać kontakt z pozostałymi przy życiu ludzmi. Nie udało się jednak, mimo że masz za sobą długą drogę. Na domiar złego okazało się, że Ziemi grozi kolejny wybuch. Nie ma już ani chwili do stracenia!

Tak więc „Ostatni V8" — to gra o życie, to szalony wyscig z czasem, to wspaniała jazda samochodem, pełna niebezpieczenstw, gdzie każdy zakręt musisz pokonać z nieprawdopodobną wprost szybkością!

Ekran podzielony jest na dwa okna. Dolne przedstawia deskę rozdzielczą Twojego samochodu, podczas gdy w gornym oknie przedstawiony jest widok z lotu ptaka na Twoj samochód i okolicę. Zegary na desce rozdzielczej informują o szybkości z jaką jedziesz, odległości do bazy oraz o czasie jaki pozostał do wybuchu bomby nuklearnej.

Skoncentruj się jednak na drodze. Jeśli tylko będziesz się jej trzymał — wszystko będzie w porządku. Pamiętaj, że jadąc z szybkością 410 km/h nie uda ci się pokonać zakrętu. Jeden wypadek i po wszystkim — koniec gry. Jeśli zaś zwolnisz możesz nie zdążyć powrócić na czas do bazy.

Gra jest wyjątkowo ekscytująca, wywołuje duże napięcie, tym bardziej, że głos w słuchawce wciąż powtarza: „V8 RETURN TO BASE IMMEDIATELY!" — V8 wracaj natychmiast do bazy!

Gra ta jest dostępna w Polsce na mikrokomputerach: Atari, Commodore 64 oraz wersja rozszerzona na Commodore 128.

Życzymy, mimo wszystko, miłej (!) zabawy.

(tb)

CHIMERA



Jakiś wrogi i groźny statek kosmiczny orbituje wokół Ziemi. Ale jest ktoś, kto go zauważył i teraz musi dowiedzieć się kim są przybysze oraz uniemożliwić ich niewątpliwie wrogie, w stosunku do nas Ziemian, zamiary.

Obcy statek kosmiczny musi zostać przez niego zniszczony — jest to bardzo niebezpieczna misja. Istnieją tylko dwie możliwości: zniszczy wroga albo zginie sam. Wyrusza więc w kierunku statku. Zbliża się coraz bardziej, lecz o dziwo, nikt do niego nie strzela. Czyżby jakiś podstęp? Bez przeszkód dostaje się na pokład, zatrzaskują się za nim drzwi i stało się — odwrotu już nie ma, musi pokonać strach i wypełnić to trudne i niebezpieczne zadanie. Jest w ogromnym pomieszczeniu, zupełnie pustym. Widzi wyjście w lewo, w prawo oraz na wprost. Najważniejszą sprawą jest teraz znalezienie klucza od odkręcania śrub — bez niego nie ma sensu ruszac w głąb statku.

Droga jest długa (4 etapy) i tylko pozornie prosta. Chociaż statek wydaje się całkowicie opuszczony przez załogę i nikt nie będzie do nikogo strzelał, to istnieje inne niebezpieczeństwo: są pewne pomieszczenia, do których nie wolno wejść — tam czeka śmierć. Wrogiem staje się również czas. Maleją zapasy wody i pożywienia. Muszą więc być uzupełniane.

Przedmioty ukryte są zawsze za murem, nie widać ich, więc trzeba wyjątkowo dokładnie badać pomieszczenie za pomieszczeniem.

W dole ekranu ukazują się informacje określające aktualny stan gry — ilość wody, pożywienia.

Trójwymiarowa grafika podobna do Alien 8 i Knightlore oraz atmosfera tajemnicy sprawiają, że gra Chimera daje dużo przyjemności, choć nie jest aż tak wyszukana jak jej znamienici bracia.

(Gramy w nią na mikrokomputerach Spectrum, Commodore i Atari!)

(tb)

LISTY —DO— LISTY

Jestem stałym czytelnikiem waszego pisma. Podczas lektury trzeciego numeru „Bajtka" nasunęły mi się pewne refleksje. Przede wszystkim Bajtkowa Lista Przebojów! Nie można wytypować „złotej dziesiątki" skoro co miesiąc pojawiają się nowe programy. Moim zdaniem należy rozszerzyc notowanie do 30 pozycji, można będzie wtedy przewidzieć pierwszą dziesiątkę, bowiem będzie wiadomo jakie gry mają szansę iść „do góry".

Andrzej Mielcarek, lat 15, 5
ul. Łanowa 21/33
87-800 Włocławek

Dużym zaskoczeniem było dla mnie pierwsze miejsce na liście przebojów gier komputerowych gry JUMPING JACK. Ja osobiście nie widzę nic w tej grze i nie gram już w nią długo.

Maciek, lat 15
Dzierżoniów

Drogi Sławku!
Mam dziewięć lat i oglądałam już program Spectrum. Najbardziej zainteresowały mnie gry komputerowe. Toteż w „Bajtku" najpierw szukam strony 18. Liczę na to, że będę następczynią Królowej Gier Komputerowych.

Anna
Warszawa



KAROL MAY I KOMPUTERY

Wojtek Serzysko, l. 15, uczen 1 klasy Zespołu Szkoł Łącznosci w Gdansku, nagrodzony w konkursie Bajtkowej Listy Przebojow.

Zainteresowania: komputery, pływanie i Karol May.

— Pytanie podstawowe: czy masz komputer?

— Nie mam.

— A więc kiedy i gdzie zetknąłeś się z nim po raz pierwszy?

— Chodziłem do siodmej klasy. Kolega dostał komputer, pokazał mi parę instrukcji, trochę pograliśmy i tak się zaczęło. Obecnie, w technikum, chodzę na dodatkowe zajęcia z informatyki.

— Czy gry nie nudzą się?

— Gdybym miał komputer w domu może już by mnie znudziły. W każdej grze chodzi w sumie prawie o to samo, chociaż grafika jest zwykle inna.

— Oprocz komputerow, co lubisz robic?

— Lubię pływac i czytac książki, najbardziej Karola Maya.

(eg)

MERITUM I — OCHRONA PROGRAMÓW

We współczesnej informatyce szeroko stosowana jest ochrona programów przed niepowołanym użytkownikiem oraz bezprawnym kopiowaniem. Artykuł ten przedstawia sposób zabezpieczenia przed kopiowaniem programów napisanych w BASIC-u i kodzie maszynowym na komputerze Meritum I. Jedynym dodatkowym „narzędziem" jest program „Dumper" dostarczony przez ZUK MERA — ELZAB, umożliwiający zapisywanie zawartości dowolnych obszarów pamięci na taśmie magnetofonowej.

Program zabezpieczony przed kopiowaniem musi spełniać dwa warunki:

1. Powinien działać bez przerw, uniemożliwiając przejście do poziomu> **READY** interpretera BASIC-a lub trybu **SYSTEM**.
2. Automatycznie rozpoczynać działanie po wprowadzeniu z tasmy.

Przejscie do poziomu> READY następuje w wyniku przerwania niemaskowalnego NMI, wciśnięcia klawisza BREAK lub napotkania przez interpreter błędu w programie. Przerwanie NMI można zablokować na dwa sposoby. Pierwszy polega na wpisaniu kodu rozkazu RET (201 dziesiętnie) do komórki pamięci o adresie 16398. Normalnie w komórce o tym adresie znajduje się kod rozkazu JP (195), a w komorkach 16399 i 16340 adres obsługi NMI. Sposób drugi wykorzystuje właściwości sprzętowe Meritum I i polega na zablokowaniu tego kanału timera 8253, na ktory podany jest sygnał NMI z klawiatury. Efekt ten można uzyskać przez wykonanie instrukcji OUT 250, 182. Przerwanie BREAK blokuje się wpisując do komórki o adresie 16396 kod rozkazu DEC A (61), a do komórki 16397 kod rozkazu RET (201). Chroniony program nie zawiera oczywiscie błędów syntaktycznych, lecz istnieje możliwość przerwania jego działania poprzez spowodowanie błędów wykonania (np. dzielenie przez zero). Sygnalizację błędow i związane z tym przerwanie działania programu likwiduje się przez wykonanie instrukcji POKE 16806, 195, POKE 16807, 127 i POKE 16808, 255.

Automatyczny start programu w kodzie maszynowym, po wprowadzeniu z taśmy, zapewnia wpisanie do komórek pamięci o adresach 16866 — 16868 kodu rozkazu JP ADR. ADR jest adresem pierwszej komórki programu, ktory ma byc wykonany. Programy napisane w Basicu muszą być poprzedzone krótkim programem w kodzie maszynowym, w związku z czym początek obszaru BASIC-a zostanie przesunięty do 17138.

W celu uzyskania wersji programu w kodzie maszynowym, nieprzerywalnego i startującego automatycznie po wczytaniu, należy program napisany w assemblerze poprzedzic następującą sekwencją dyrektyw i rozkazów:

```
ORG     16866
DB      195
DW      ADR
ORG     ADR
LD      A, 182
OUT     (250),A
```

Przepis dla programow napisanych w BASIC-u jest bardziej złożony.

1. Załadować z taśmy DUMPER.
2. Załadować z tasmy program lub umieścić go w pamięci za pomocą instrukcji POKE:

```
33, 164, 64     LD      HL, 16548
17, 242, 66     LD      DE, 17138
115             LD      (HL), E
35              INC     HL
114             LD      (HL),D
27              DEC     DE
62,0            LD      A, 0
18              LD      (DE), A
205, 74, 27     CALL    6986
201             RET
```

3. Uruchomić powyższy program przez PRINT USR (0)
4. Zablokować BREAK, NMI i obsługę błędow
5. Od adresu 17129 wpisać program

```
17, 30,29       LD      De, 7454
213             PUSH    DE
195, 93 27      JP      7005
```

6. POKE 16866, 195. POKE 16867, 233: POKE 16868, 66 — 256 × 66 + 233 = 17129!
7. Wprowadzic z klawiatury lub taśmy magnetofonowej program w BASIC-u
8. Odczytać adres ostatniej komórki programu napisanego w BASIC-u — PRINT (PEEK/16633) + 256 × PEEK (16634)
9. Uruchomic DUMPER wykorzystując PRINT USR (0).
10. Zapisać na tasmę obszar pamięci od adresu 16396 do ostatniej komórki programu.

Sławomir Borucki
Marek Mizeracki



MINI ORGANKI

Jeżeli interesujecie się komputerami i tworzeniem własnej muzyki to możecie spróbować swych sił twórczych po wczytaniu poniższego programu do mikrokomputera Meritum.

Program ten przeistacza klawiaturę mikrokomputera w klawiaturę „Mini Organów". Przy pomocy klawiszy od 0 do 7 możecie tworzyć własne melodie, może trochę odbiegające od światowych standartów brzmieniowych, ale to jest tylko mikrokomputer.

I co? — mikrokomputery potrafią wszystko, no — prawie wszystko.

Bogdan A.Grzybowski

```
0 REM !!        MINI ORGANKI          !!
1 REM !! BOGDAN A. GRZYBOWSKI !!
2 REM !!      I N F O R M I K        !!
10 DATA 58,16,56,254,0,40,249
20 DATA 211,254,61,32,251,24,242
30 FOR A=32000 TO 32013
40 READ B
50 POKE A,B
60 C=C+B
70 NEXT A
80 IF C<>1748 THEN PRINT"ZŁE DANE"
90 POKE 16526,0:POKE 16527,124
100 U=USR(0)
```

JAK MASZ NA IMIĘ

Mikrokomputer MERITUM posiada cztery tryby pracy. Jednym z nich to tryb systemowy. W trybie tym możliwe jest ładowanie programów w języku maszynowym. Wprowadzanie programów jest o tyle utrudnione, że należy podac poprawną nazwę. Aby wprowadzić program o niewiadomej dla nas nazwie, mozemy posłużyc się poniższym programem. Po uruchomieniu rozkazem RUN, program ładujący wywołujemy komendą LOAD. Działanie programu: po wprowadzeniu rozbiegówki komputer wypisuje nazwę programu, w prawym, dolnym rogu ekranu, po czym wprowadza program.

Robert Braksator
INFORMIK

```
10 REM PROGRAM ŁADUJACY  BRAX  INFORMIK
20 FOR A=16448 TO 16472:READ B
30 POKE A,B:NEXT A
40 DATA 205,147,2,205,53,2,254,85,32,249,6,6,33
50 DATA 250,63,205,53,2,119,35,16,249,195,231,2
60 REM DEKLARACJA INSTRUKCJI LOAD
70 POKE 16773,64:POKE 16778,64
80 NEW
```

MIGAJĄCY KURSOR

Wiele seryjnie produkowanych mikrokomputerów przez czołowe firmy światowe posiada tak zwany migający kursor, czyli zwykły kursor lecz migający z pewną częstotliwością.

Kursor taki stosowany jest w celu łatwiejszej lokalizacji miejsca pracy na ekranie (wpisywanie tekstu, edycja, itp.) Jest on zrealizowany na drodze czysto programowej, toteż znając dowolny system mikrokomputerowy możemy ułożyć program realizujący funkcje migającego kursora.
Poniższy program przeznaczony jest na mikrokomputer Meritum 16K i Meritum 48K, a realizuje opisaną funkcję.

```
0 REM !!   MIGAJACY KURSOR    !!
1 REM !! BOGDAN A. GRZYBOWSKI !!
2 REM !!   I N F O R M I K    !!
10 DATA 58,159,64,206,0,50,159,64,48,8,237
20 DATA 75,32,64,10,238,144,2,195,227,3
30 FOR A=32739 TO 32759
40 READ B
50 C=C+B
60 POKE A,B
70 NEXT A
80 IF C<>2043 THEN PRINT"ZŁE DANE"
90 POKE 16407,127
```

Program w BASIC-u służy do załadowania procedury maszynowej (linie 30,40,60,70) sprawdzenia poprawności zapisanych wartości w instrukcji DATA (linie 50,80) oraz uruchomienia programu maszynowego (linia 90).
Program realizujący funkcje migającego kursora realizowany jest z poziomu języka wewnętrznego, a wygląda następująco:

```
       LD A,(409FH)     ; POBIERZ WARTOSC LICZNIKA
       ADC A,00H        ; A=A+y+CY
       LD (409FH),A     ; PRZECHOWAJ NOWA WARTOSC A
       JR NC, KLAW      ; SKOCZ JESLI CARRY=0
       LD BC,(4020H)    ; POBIERZ POZYCJE KURSORA
       LD A,(BC)        ; POBIERZ KSZTAŁT KURSORA
       XOR 90H          ; SUMA MODULO DWA
       LD (BC),A        ; WYPISZ NOWY KURSOR
KLAW:  JP 03[illegible]H         ; SKOK POWROTNY
```

Poprawnie zapisany program powinien działać po uruchomieniu go zleceniem RUN. Jeżeli jednak chcemy wyłączyć migający kursor to użyjemy zlecenia:
POKE 16407,3
Dla ponownego włączenia naszego kursora użyjemy zlecenia
POKE 16407,127
Na koniec jeszcze jedna możliwość, a mianowicie jeżeli chcemy zmienić kształt i częstotliwość kursora to użyjemy zleceń
POKE 32743,częstotliwość
POKE 32755,znak
Życzę ciekawych efektów

Bogdan A. Grzybowski
INFORMIK

BIORYTMY

Jeśli czujesz się fatalnie, łamie cię w kościach a głowa spuchła ci jak bania, to z pewnością dlatego, że krzywa fizyczna twojego cyklu biologicznego przecięła się z osią.

Pokłóciłeś się z dziewczyną i pobiłeś kolegę z pracy — to emocjonalny dzień krytyczny. Dostałeś cztery lufy w ciągu trzech lekcji, wiadomo, kryzys intelektualny. To więcej niż pewne! Chyba, że w pierwszym przypadku łapie cię grypa, w drugim — mucha usiadła ci na nosie, a w trzecim — bez reszty pochłonęło cię błogie lenistwo.

Co by jednak nie mówić o biorytmach — niezależnie, czy się w nie wierzy, czy też nie — mają one jedną nieocenioną zaletę: niezmiernie rzadko zdarza się, że jesteśmy sprawni pod każdym względem. A więc gdy coś nam nie wyjdzie, np. oblejemy egzamin, wystarczy obliczyć sobie biorytm i mamy już spokojne sumienie. Bo przecież, choć intelektualnie stoimy zupełnie nieźle, to fizycznie ... lepiej nie mówić. I w dodatku emocjonalnie też spadamy. Wszystko przez te nerwy!

Myślę, że przekonałem najbardziej nawet sceptycznych czytelników o użyteczności programu do obliczania biorytmów, tym bardziej, że jest to program, który może działać na każdym mikrokomputerze. Jedynie posiadacze Spectrum będą musieli zmienić wszystkie oznaczenia zmiennych łańcuchowych na jednoliterowe, ale to przecież drobnostka.

Działanie programu jest proste i nie wymaga specjalnego komentarza. Tym bardziej, że program komentuje się sam, poprzez oznaczenia zmiennych.

Najpierw komputer liczy ilość dni, które przeżył badany, następnie oblicza reszty z dzielenia ilości dni przez długości cykli i na tej podstawie określa jego fizyczną, intelektualną i emocjonalną kondycję w danym dniu. Teraz tylko dobrać odpowiedni komentarz i sprawdzamy co będzie jutro.

A tak przedstawiała się moja sytuacja w dniu pisania tego programu. Jeśli więc coś będzie nie tak, sami widzicie, że nie jestem w najlepszej formie psychicznej...

Roman Poznański

```
 15 . 9 . 1986 r.
Sprawnośc fizyczna maksymalna ( 99 %). Tendencja rosnąca.
Sprawnośc psychiczna słaba ( 22 %). Tendencja rosnąca.
Sprawnośc intelektualna niezła ( 91 %). Tendencja malejąca.
```

```
1 REM ********************************
2 REM ********** biorytmy ************
3 REM ******** uśmiechnij się ********
4 REM ********************************

99 REM * deklaracje zmiennych i tablic *

100 DIM sprawnosc(3)
110 DIM cykl(3)
120  FOR i=1 TO 3
130   READ cykl(i)
140  NEXT i
150 DIM sprawnosc$(3)
160  FOR i=1 TO 3
170   READ sprawnosc$(i)
180  NEXT i
190 DIM miesiac(12)
200  FOR i=1 TO 12
210   READ d
220   LET miesiac(i)=30+d
230  NEXT i

499 REM ************ dane ************

500 DATA 23, 28, 33
510 DATA "fizyczna", "psychiczna", "inte
lektualna"
520 DATA 1,-2,1,0,1,0,1,1,0,1,0,1

999 REM ****** komputer pyta się ******

1000 PRINT "Podaj datę urodzenia"
1010 INPUT "dzien";dzien0
1020 INPUT "miesiac";miesiac0
1030 INPUT "rok";rok0
1040 PRINT "Podaj datę, ktora cię intere
suje"
1050 INPUT "dzien";dzien1
1060 INPUT "miesiac";miesiac1
1070 INPUT "rok";rok1
1999 REM ***** ile dni przeżyłeś? *****
2000 LET rok=rok0
2010 GOSUB 5010
2020 LET liczbadni=miesiac(miesiac0)-dzi
en0+1
2030  FOR i=miesiac0+1 TO 12
2040   LET liczbadni=liczbadni+miesiac(i
)
2050  NEXT i
2060  FOR i=rok0+1 TO rok1-1
2070   LET liczbadni=liczbadni+365
2080   IF i/4=INT(i/4) THEN LET liczbadn
i=liczbadni+1
2090  NEXT i
2100 GOSUB 5000
2110  FOR i=1 TO miesiac1-1
2120   LET liczbadni=liczbadni+miesiac(i
)
2130  NEXT i
2140 LET liczbadni=liczbadni+dzien1

2998 REM komputer liczy twoja sprawność
2999 REM ******** i odpowiada *********

3000 PRINT dzien1;".";miesiac1;".";rok1;
"r."
3010  FOR i=1 TO 3
3020 LET reszta=liczbadni-INT(liczbadni/
cykl(i))*cykl(i)
3030   LET sprawnosc(i)=INT(100*SIN(resz
ta*3.14/cykl(i)))
3040   LET ocena$="maksymalna"
3050   IF sprawnosc(i)<95 THEN LET ocena
$="niezła"

3060   IF sprawnosc(i)<60 THEN LET ocena
$="średnia"
3070   IF sprawnosc(i)<30 THEN LET ocena
$="słaba"
3080   IF sprawnosc(i)<15 THEN LET ocena
$="bardzo zła"
3090   IF sprawnosc(i)=0  THEN LET ocena
$="fatalna - dzień krytyczny"
3100 LET tendencja$="malejaca"
3110 IF reszta<cykl(i)/2 THEN LET tenden
cja$="rosnaca"
3120   PRINT "Sprawnosc ";sprawnosc$(i);
" ";ocena$;" (";sprawnosc(i);"%). Tenden
cja ";tendencja$;"."
3130  NEXT i
3140 INPUT "naciśnij >ENTER<";e$
3150 LET liczbadni=liczbadni+1
3160 LET dzien1=dzien1+1
3170 IF dzien1<=miesiac(miesiac1) THEN G
OTO 3000
3180 LET dzien1=1
3190 LET miesiac1=miesiac1+1
3200 IF miesiac1<=12 THEN 3000
3210 LET rok1=rok1+1
3220 GOSUB 5000
3230 GOTO 3000

4999 REM *** czy rok jest przestepny ***

5000   LET rok=rok1
5010   LET miesiac(2)=28
5020   IF rok/4=INT(rok/4) THEN miesiac(
2)=29
5030   RETURN
```

GIEŁDA *(ceny na dzień 15.X.1986 r.)*

		GIEŁDA BAJTKA (tys zł)	KOMIS (tys zł)	AUSTRIA (średnie) (oS)	FRANCJA (średnie) (FF)	RFN (średnie) (DM)	WLK BRYT (średnie) (£)
SINCLAIR	ZX 81	25–35	—	580	—	79	—
	ZX Spectrum 48 kB	75–85	100	1400	—	150–250	45–65
	ZX Spectrum Plus	110–125	140–160	1600	1350	200–300	65–80
	ZX Spectrum 128 kB	190	250	—	2200	500–530	120
	Drukarka SEIKOSHA GP50S	80–90	110	—	—	248	60–65
	Interface Kempston	5–13	20	250	300	35	6–9
	Stacja dyskietek OPUS 1(3.5")	—	—	500	—	395	95–105
COMMODORE	C–64	150–165	160–180	3500	1900	370–449	90–110
	C–128	285–310	430	6500	2890	650–700	210–230
	C–128 D	580	—	2500	6850	1250	390–410
	Magnetofon 1531	30–35	40	900	350	70	25
	Stacja dyskietek 1541	155–170	200	4500	1950	470	110–150
	Stacja dyskietek 1570	—	300	6900	2300	490–540	160
	Drukarka MPS 801	—	150	2900	2200	199	—
	Drukarka MPS 803	160	200	—	2800	300–330	140–150
	Dysk[illegible] 5 1/4 (srednia jak)	0.6–1.2	1–2	10–25	7	0.5–3	0.8–2
ATARI	800 XL	70–85	90–100	1500	900	140–180	60
	130 XE	130–140	—	2100	1400	360	150
	Stacja dyskietek 1050	130–140	160–200	2200	2150	379	130
	Magnetofon	30–35	30–35	800	—	69	20
	Drukarka 1029	—	200	1990	—	—	85
AMSTRAD	464 z monit monochromat	210	300	6500	2690	570	165–182
	464 z monit kolor	310	350–400	9500	3990	1050	249
	6128 z mon[illegible] monochromat	410	700–800	11000	3390	998	250
	6128 z monit kolor	—	1 mln	15800	5290	1500	329
	PCW 8256	—	1.3 mln	—	5920	1595	345
	D[illegible]skietki 3"	4–5	6	—	35	8–14	3.5–4
	Stac[illegible]a [illegible]ek 3" do 464	—	295	6000	260	549	140

Niezbyt zasobni w gotówkę francuscy miłośnicy mikroinformatyki mają do dyspozycji rynek używanego ale całkowicie sprawnego sprzętu mikrokomputerowego, na którym trafiają się prawdziwe okazje. I tak np. w październiku można było kupić następujące zestawy: CPC 6128 z monitorem zielonym i 40 dyskietkami za 3500 franków (wartość 5400), CPC 464 z napędem dysków DD1, drążkiem sterowym, piórem optycznym, monitorem kolorowym, programami i książkami za 5000 Fr (wartość 7000), ATARI 130XE z magnetofonem, sprzęgiem do TV, drążkami sterowymi i kilkoma programami za 1800 Fr (wartość 2300), COMMODORE 64 z wyjściem SECAM, napędem dysków 1541, monitorem czarno-białym i programami za 3500 Fr (wartość ponad 5500), Spectrum+ ze sprzęgiem wideo, magnetofonem, książkami i grami za 1100 Fr (wartość ponad 1600 Fr).



GIEŁDA BAJTKA

Giełda Bajtka odbywa się w każdą sobotę w Warszawie, w Szkole Podstawowej nr 25, przy ul. Grzybowskiej 35 (róg Marchlewskiego) w godz. 14.00–19.00. Frekwencja — jak widać — dopisuje. Zapraszamy na świąteczne zakupy!



KONKURS O ZŁOTĄ DYSKIETKĘ „BAJTKA"



SAMI O SOBIE

MERA ZAP

Jestesmy klubem komputerowym działającym przy Zespole Szkoł Technicznych MERA-ZAP w Ostrowiu Wielkopolskim. Istniejemy już trzeci rok, od niedawna zas należymy do Ogólnopolskiej Federacji Klubow Komputerowych Młodych Mistrzow Techniki.

Nasza szkoła kształci około 1500 uczniow, z których większość znajduje następnie pracę w naszym zakładzie. Mamy własne studio telewizyjne, filmotekę, dobrze wyposażone pracownie. Ale dla nas, członków klubu, a chyba i dla dyrektora szkoły Waldemara Gostomczyka, najważniejszą sprawą jest informatyka i komputery

Jest nas przeszło stu, głownie uczniów z klas o profilu „elektroniczne maszyny i systemy cyfrowe". Choć każdego dnia nasi członkowie mają do swojej dyspozycji komputery przez 4 godziny, jednak bardzo trudno „dopchać" się do sprzętu, a chętnych ciągle przybywa. Niedawno zakończylismy prace przy budowie sieci komputerow „MERITUM". Stworzylismy ją na bazie mikrokomputera „MERA" i interface'u, za który otrzymalismy nagrodę na eliminacjach wojewódzkich TMMT. Posiadamy kilkadziesiąt mikrokomputerow, w tym SPECTRUM, MERITUM, COMMODORE, ZX 81 oraz komputer profesjonalny IBM-PC z pamięcią 25 MB, ktory ufundowały nam w zeszłym roku opiekuncze zakłady. W najbliższym czasie chcemy, aby każdy nasz absolwent znał i umiał się posługiwac językiem BASIC i różnymi typami komputerow.

Największym problemem, nie tylko zresztą dla nauczycieli, lecz i dla nas uczniow, jest brak wystarczającej ilosci programow dydaktycznych. Stworzenie takiego banku programów jest potrzebą chwili.

Własnym dorobkiem klubu jest stworzenie kilku programow z zakresu fizyki, matematyki oraz elektroniki. Gromadzimy systematycznie wszelką dostępną literaturę z informatyki i już przystąpilismy do skomputeryzowania naszej biblioteki. Informatyka to jednak nie tylko komputery. W szkole wykonujemy zegary elektroniczne, zamki szyfrowe, systemy alarmowe oraz interfejsy do mikrokomputerow. Staramy się także propagowac wiedzę informatyczną wsrod naszych rowieśników spoza szkoły. Podczas tegorocznych wakacji członkowie naszego klubu prowadzili pokazy sprzętu komputerowego na obozach w NRD oraz w Boszkowie.

MERA ZAP

Zgodnie z zapowiedziami publikujemy w tym numerze kupon zgłoszenia do konkursu klubow mikrokomputerowych o ZŁOTĄ DYSKIETKĘ „BAJTKA"

We wspołzawodnictwie mogą brać udział wszystkie kluby mikrokomputerowe: szkolne zakładowe, działające pod patronatem organizacji młodzieżowych, domow kultury itd. Zgłoszenia mogą przysyłac także kluby „prywatne"

Pierwszym, najważniejszym celem konkursu jest stworzenie możliwosci wymiany doswiadczen pomiędzy klubami, popularyzacja najciekawszych pomysłów dotyczących organizacji pracy w klubie i jego działalności społecznej. Chcemy pokazac te najciekawsze formy działalnosci i tym samym dać bogatszą możliwosc wyboru kierunku pracy nowym klubom, a także wzbogacic formy działania klubow o dłuższym stażu.

Wypełnione kupony zgłoszenia do konkursu prosimy przysyłac do 28 lutego 1987 roku. Do kuponu należy dołączyć:

— Zasady działania klubu (statut)
— Krotki opis dotychczasowej działalnosci i osiągnięc.
— Zamierzenia i plany na rok 1987

Po otrzymaniu zgłoszen redakcja „Bajtka" wpolnie ze wspołorganizatorami konkursu (Minister ds. Młodzieży, Związek Młodzieży Wiejskiej, Związek Harcerstwa Polskiego, Centralne Biuro TMMT, Centralna Składnica Harcerska) zapozna się bliżej z pracą wytypowanych klubow i w maju 1987 roku opublikujemy wyniki konkursu. Na zwycięzcow czekają cenne nagrody, przede wszystkim sprzęt komputerowy.

Czekamy na zgłoszenia!

„Bajtek"
ul. Wspolna 61
00-687 Warszawa

KUPON ZGŁOSZENIA DO KONKURSU O ZŁOTĄ DYSKIETKĘ „BAJTKA"

Nazwa klubu.

Adres: ul. kod: miejscowosc:

Numer telefonu (klubu, prezesa, jednego z członkow):

Nasz klub działa od roku, obecnie zrzesza . członkow

Opiekun (szkoła, zakład pracy, organizacja patronacka itp)

Zgłaszamy nasz klub do wspołzawodnictwa O ZŁOTĄ DYSKIETKĘ „BAJTKA". W załączeniu przesyłamy:
— Zasady działania naszego klubu
— Opis dotychczasowej działalnosci
— Zamierzenia i plany na rok 1987.

Opiekun: Prezes klubu

Adres (telefon):

(podpis i pieczątka) podpis:

DLACZEGO IBM?

Znaczna część reklam profesjonalnych, szesnastobitowych mikrokomputerów zawiera następujące sformułowanie: nasz produkt jest zgodny programowo z IBM PC. Praktycznie takie stwierdzenie oznacza zwykle, że pod względem funkcjonalnym produkt jest prawie idealną kopią IBM-a.

Jest to jedno z wielu potwierdzeń faktu, że IBM PC stał się nieoficjalnym standardem. Czy jest nim zasłużenie? Bardziej ambitni producenci twierdzą, że nie i probują lansować własne konstrukcje. Niezależnie od tego kto ma rację warto chyba zapoznać się ze sprzętem, który (w oryginale lub kopiach) jest chyba najpopularniejszym profesjonalnym mikrokomputerem na świecie.

■ DRZEMIĄCY KOLOS

Zanim jednak zajmiemy się szczegółami technicznymi cofnijmy się w czasie o około pięć lat. Przy dzisiejszym tempie rozwoju informatyki, pięć lat to już tak zamierzchła przeszłość, że coraz trudniej odróżnić fakty od legendy ale spróbujmy zobaczyć jak to się zaczęło.

W momencie pojawienia się na rynku pierwszych mikrokomputerów firma IBM zachowywała się na pozór biernie. Wiele osób uważa, że było to działanie z pełną premedytacją — niech inni sondują rynek i popełniają kosztowne pomyłki. Być może to prawda, bo gdy się już wydawało, że gigant zaspał lub, że „odpuścił" rynek mikro, pojawił się właśnie IBM PC. Pojawił się w sposób niezbyt typowy. Wbrew swoim zwyczajom firma zmontowała go z podzespołów innych producentów. Wbrew zdrowemu rozsądkowi (tak się wtedy wydawało) nie ukrywała szczegółów technicznych (prawem autorskim chroniona jest tylko zapisana w ROM-ie podstawowa część systemu operacyjnego). W sumie nie było to nic innego jak zaproszenie do kopiowania, które zostało z ochotą przyjęte przez wielu producentów sprzętu. Dzięki temu liczba oferowanych egzemplarzy nowego komputera szybko rosła, na co natychmiast odpowiedzieli producenci oprogramowania. Wkrótce było już tyle wspaniałych programów, że ludzie chcieli kupować tylko te komputery, które pozwalały korzystać z tych wspaniałości. Wzrastająca liczba instalacji stymulowała produkcję nowych, jeszcze wspanialszych programow i w ten sposób kółko się zamyka. Być może morał z tej historyjki jest taki, że życzliwym łatwiej w życiu niż tym, którzy zazdrośnie strzegą swoich tajemnic.

■ OD ŚRODKA

Ale dosć żartów, przyjrzyjmy się budowie PC, a przekonamy się, że rzeczywiście może to być komputer dla prawie każdego. Podstawowym składnikiem systemu jest tzw. płyta systemowa (ang. mother board), zawierająca mikroprocesor, pamięci RAM i ROM, i właściwie nic więcej oprócz kilku dziur, czyli miejsc przeznaczonych na tzw. karty rozszerzeń systemu. Karty te, to zwykłe płytki drukowane, z elektroniką realizującą dowolne, wybrane przez użytkownika funkcje. Za chwilę omówimy to dokładniej na przykładzie konkretnego modelu, najpierw jednak zajmijmy się przez chwilę urządzeniem, które właściwie może pracować z każdym komputerem, ale „pod strzechy" trafiło wraz z mikrokomputerami profesjonalnymi.



▲ IBM AT

IBM XT ▶

■ PAMIĘĆ NA TWARDO

Mam na myśli popularny WINCHESTER, albo jak kto woli „twardy dysk", „sztywny dysk", „dysk stały" (ang. hard disk, fixed disk). Działa on na takiej samej zasadzie jak stacje dysków elastycznych (dyskietek) czy stacje dysków w dużych komputerach. Pokryty substancją magnetyczną dysk wiruje wokół własnej osi a nad nim umieszczone są głowice zapisujące i odczytujące dane, trochę podobne do głowic magnetofonowych. Jednak w przeciwieństwie do dyskietek, których możemy mieć wiele i wymieniać dowolnie często, nośnik magnetyczny wewnątrz winchestera jest na stałe hermetycznie zamknięty. Dzięki temu urządzenie może być dużo szybsze i precyzyjniejsze niż stacja dyskietek. Pojemność pierwszych popularnych dysków tego typu wynosiła 10MB (megabajtów), czyli w przybliżeniu 5 tysięcy stron maszynopisu! Przewidywany przez producenta okres eksploatacji — 5 lat! Dodam jeszcze, że dziś „dziesiątki" spotyka się coraz rzadziej. Oferowane dyski mają pojemności kilka lub nawet kilkanaście razy większe.

Dzięki swojej ogromnej pojemności twardy dysk znacznie usprawnia prace — najczęściej używane programy i dane mamy „w komputerze" na stałe, co eliminuje potrzebę częstego przekładania dyskietek. Jego użycie otwiera również nowe dziedziny zastosowań, wymagające przetwarzania wielkiej liczby danych.

■ PC I RODZEŃSTWO

Wróćmy teraz do pierwszego **IBM PC**, który jeszcze nie miał twardego dysku. Obudowa systemu mieściła całą elektronikę (płytę głowną i karty rozszerzeń), zasilacz i dwie stacje dyskietek 5.25 cala o pojemności 320KB (kilobajtów) każda. Do tego dołączamy jeszcze monitor — IBM PC w zasadzie nie współpracuje z telewizorem — i klawiaturę, zawierającą 83 klawisze w charakterystycznym, zaakceptowanym przez większość producentów układzie.

Następcą tego modelu był **IBM PC XT**, ktorym zajmiemy się szerzej, bo jest w tej chwili najpopularniejszy w Polsce. W leżącej na dnie panelu płycie systemowej jest miejsce na 8 stojących pionowo kart. Mikroprocesor ten sam co w zwykłym PC — Intel 8088 (czasami można spotkać Intel 8086), podstawowa pamięć RAM 256 KB. Pamięć ROM o pojemności 40KB zawiera interpreter BASIC-a, samoczynny test (ang. self-test), czyli program, który w momencie włączenia komputera do sieci testuje wszystkie podzespoły elektroniczne, oraz wspomnianą już część systemu operacyjnego, tzw. BIOS (ang. Basic Input/Output System, nie mylić z językiem BASIC). Resztę przestrzeni zajmuje zasilacz z wentylatorem, chłodzącym całe urządzenie, oraz miejsce na pamięci magnetyczne. Miejsce to jest wykorzystane inaczej niż w poprzednim modelu. Zastosowane stacje dysków elastycznych, mimo nieco większej pojemności (360K), są o połowę niższe, co pozwala umieścić w tej samej przestrzeni nie dwa, a cztery napędy. Tylko po co komu aż cztery, tym bardziej, że dokładnie tyle samo miejsca co stacja dyskietek zajmuje kompletny twardy dysk, dający dużo większe możliwości. Typowy zestaw



to winchester, jeden lub dwa napędy dyskietek. Czwarte, wolne miejsce, zasłonięte plastikową płytką, można zapełnić za rok czy dwa, gdy np. okaże się, że jednak potrzebujemy dwóch stałych dysków. Firma IBM głosi zasadę „Twój komputer może rosnąć razem z Tobą".

Oczywiście, wstawiając stację dyskietek, czy twardy dysk trzeba dołożyć kartę zapewniającą ich współpracę z procesorem, tak zwany „kontroler". To samo dotyczy do-



UWAGA STRES !!!

Czego to ludzie nie wymyślą! IBM wymyślił kartę ... do badania stresów.

Karta ma wymiar prostokąta o długości boków 5 na 7 centymetrów. Pośrodku niej znajduje się czarny kwadracik o powierzchni 4 centymetrów kwadratowych, kryjący w sobie substancję termoczułą, reagującą na najmniejsze zmiany temperatury (podobnie jak dostępny w naszym handlu, zakładany na czoło termometr-opaska).

Pod wpływem temperatury przyłożonego do karty na 10 sekund kciuka oraz pod warunkiem, ze temperatura otoczenia nie jest mniejsza niż 20 stopni Celsjusza, kwadracik może przybierać jeden z czterech kolorów, oznaczających odpowiednio czy badany osobnik jest:

- mocno zestresowany — kolor czarny
- napięty — kolor czerwony
- spokojny — kolor zielony
- zrelaksowany — kolor niebieski

W trakcie przeprowadzonego w redakcji „Bajtka" szybkiego testu, stwierdziliśmy co następuje:

Red. W. Siwiński — czerwono-zielony, całkiem nieźle jak na zastępcę redaktora naczelnego SM

łączania innych urządzeń, np. monitora. Przy okazji zwróćmy uwagę, ze parametry produkowanej przez komputer grafiki określa tylko karta i monitor. Czyli jeśli w pewnym momencie stwierdzamy, że potrzebna nam jest lepsza grafika, to nie musimy zmieniać komputera. Wystarczy wymienić kartę i ewentualnie monitor. Dokładnie tak samo możemy postępować z innymi urządzeniami.

Jeśli potrzebujemy drukarki, to dokładamy kartę z typowym równoległym złączem (ang. interface) CENTRONICS, przez które możemy ją dołączyć. Zwykle taka karta zawiera również szeregowe złącze RS-232, pozwalające dołączyć inne urządzenia, np. do teletransmisji. Miejsce nie może się marnować, więc na jedną kartę wciskamy jeszcze dodatkowo kostki rozszerzające pamięć operacyjną (górna granica 640K), i zegar z baterią, dzięki którym nasz IBM zawsze zna aktualną datę i godzinę.

Gdy nasza praca wymaga wielu obliczeń i zależy nam na szybkości, to nic prostszego jak dołożyć do komputera drugi procesor. Na płycie systemowej jest przygotowana podstawka na koprocesor arytmetyczny, czyli mieszczący się w jednym układzie scalonym, wyspecjalizowany procesor do szybkiego wykonywania operacji arytmetycznych. Najczęściej jest to Intel 8087. Jeśli tylko zastosujemy właściwe oprogramowanie przejmie on część pracy centralnego mikroprocesora i w ten sposób przyśpieszy wykonywanie programów użytkowych.

Dla tych, którzy nabrali ochoty na posiadanie IBM-a, ale obawiają się, że na tak duży komputer zabraknie im miejsca w mieszkaniu podaję orientacyjne wymiary jednostki centralnej (w centymetrach): 50 × 40 × 15.

Jeśli, mimo wszelkich rozszerzeń, nasz PC XT jest za mało pojemny lub za wolny, możemy kupić nowszy model. **IBM PC AT** na mikroprocesor Intel 80826, pamięć RAM od 512kB do 3 MB. Standardowo wyposażony jest w twardy dysk i stację dyskietek, która dzięki poczwórnej gęstości zapisu mieści się na dyskietce 5.25 cala 1.2MB danych. Dodatkowo, stacja ta potrafi również czytać dyskietki o pojemności 360KB. Oczywiście, programów, których używaliśmy na XT można używać również na AT. Przy tych wszystkich możliwościach cały mikrokomputer mieści się w niewielkim pudełku (ok. 55 × 45 × 17), które wraz z monitorem, klawiaturą i drukarką można bez trudu zmieścić na przeciętnym biurku.

■ PROGRAMY

Sam sprzęt to jeszcze nie wszystko, potrzebne jest oprogramowanie. Wraz z mikrokomputerem dostarczany jest system operacyjny DOS. Jego główne zadania to: interpretacja zleceń wydawanych przez użytkownika i obsługa zbiorów zawierających dane i programy na nośnikach magnetycznych. Zaletą DOS-u jest podobieństwo do znanych systemów operacyjnych: CP/M dla mikrokomputerów, oraz UNIX i RSX-11 dla bardzo popularnych minikomputerów. W razie potrzeby system można łatwo modyfikować, np. wzbogacając go o stworzone przez siebie nowe zlecenia. Inne dostępne systemy operacyjne, to CP/M, oraz, przeznaczony dla modelu AT, XENIX, pozwalający korzystać z jednostki centralnej kilku osobom na raz — oczywiście za pośrednictwem dodatkowo dołączonych końcówek.

Oprogramowanie użytkowe działające na IBM PC zawiera kilka tysięcy pozycji. Składają się na nie kompilatory znanych z dużych komputerów języków programowania: FORTRAN, PASCAL, COBOL itd., bazy danych, edytory tekstów, poważne programy do zarządzania przedsiębiorstwem i, na drugim biegunie, cały wachlarz gier. Dla pracowitych programy ułatwiające pisanie programów, dla mniej pracowitych narzędzia do „przyswajania" programów zabezpieczonych przed kopiowaniem, sprzedawane oczywiście „tylko do robienia zapasowych kopii programów, wyłącznie na własny użytek".

Właściwie w każdej dziedzinie zastosowań mikrokomputerów oferta jest tak wszechstronna, że trudno się jej oprzeć. I na ogół ludzie się nie opierają. Po prostu kupują IBM-a.

Andrzej Pilaszek

Red. R. Poznański — czarny..., tłumaczył się, ze mył ręce w bardzo zimnej wodzie.

Red. S. Polak — niebieski; chyba nareszcie skończył „Impossible Mission".

Red. R. Wojciechowski — czerwony; kolor miłości... do „Bajtka".

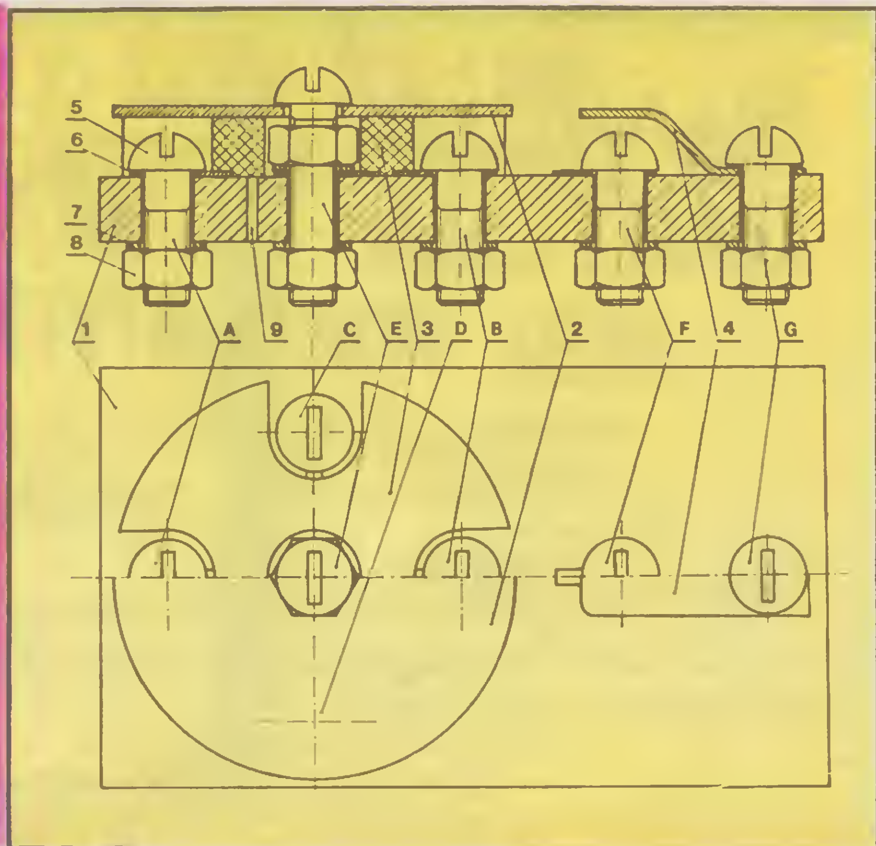
Red. W. Roszkowska — zielono-niebieski; jako grafik, stwierdziła autorytatywnie, że jest to ostatnio najmodniejszy zestaw kolorów.

T. Brojak (sekretariat) — niebiesko-czarny (sic!); zawsze lubiła muzykę, która trąci „myszką".

Dalszych testów nie będzie.

(jz)

DRĄŻEK (?) STEROWY



Rys. 1. Konstrukcja manipulatora: 1 — podstawa, 2 — blaszka, 3 — gąbka, 4 — styk sprężysty „fire", 5 — wkręty z łbami okrągłymi (7 sztuk), 6 — podkładka stykowa (7 sztuk), 7 — podkładka zwykła (8 sztuk), nakrętka ośmiokątna (8 sztuk), 9 — otwór na przewody elektryczne.

Nasz drążek sterowy spełniał będzie wszystkie funkcje tradycyjnych manipulatorów, z tą jednak różnicą, że... obywamy się bez samego drążka.

Jego rolę przejmuje okrągła blaszka z otworem (element 2 na rys. 1), która równocześnie jest głównym stykiem elektrycznym. Otwór w blaszce (2) jest większy niż średnica wkręta (5), na którym blaszka jest zamocowana. Jest on jednak mniejszy niż średnica łba wkręta. Jeśli pomiędzy blaszką i nakrętką pozostawimy trochę miejsca, uzyskamy połączenie pozwalające na przechwyt blaszki w dowolnych kierunkach a tym samym na zamykanie obwodu elektrycznego pomiędzy blaszką a czterema pozostałymi wkrętami. Rolę sprężyny podtrzymującej blaszkę spełnia płaski kawałek gąbki (3) wycięty według kształtu przedstawionego na rysunku 1.

Trudno wyobrazić sobie prostszą konstrukcję. Po zgromadzeniu odpowiednich materiałów — myślę, że nie będziecie mieli z tym żadnych kłopotów — wykonanie naszego „drążka" nie powinno nam zająć więcej niż pół godziny. Oto co będzie nam potrzebne:

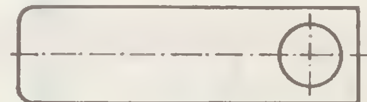
- cienka deseczka (sklejka) lub płaski kawałek tworzywa sztucznego)
- okrągła blaszka (grubości ok. 1 mm)
- kawałek gąbki (im twardsza tym lepiej)
- pasek sprężystej blachy (najlepiej stalowej)
- cienka blacha miedziana lub mosiężna
- wkręty z okrągłymi łbami (7 sztuk)
- podkładki o średnicy dobranej do średnicy wkrętów (8 sztuk)
- nakrętki sześciokątne (8 sztuk)
- wtyczka typu Eltra 881 (z tym może być trochę kłopotu)

Na rysunkach nie podałem wymiarów. Będą one zależały od wielkości elementów, jakie uda ci się znaleźć. Pracę zaczniemy od wycięcia prostokątnej deseczki i wywiercenia 7 otworów na wkręty (patrz rysunek 1) oraz otworu (9) na wyprowadzenie przewodów. Dokładnie w środku okrągłej blaszki wiercimy otwór — tak aby wkręt mieścił się w nim ze sporym luzem — mniejszy jednak od średnicy łba wkręta.

Z gąbki wycinamy kółko o średnicy okrągłej blaszki. Następnie wykonujemy podcięcia na śruby według rysunku 1. Grubość gąbki powinna być tak dobrana, by po zmontowaniu manipulatora gąbka była wstępnie ściśnięta.



Rys. 2. Podkładka stykowa.



Rys. 3. Blaszka przycisku „fire"



Rys. 4. Wtyczka Eltra 881, widok od strony połączeń lutowniczych.

Mosiężna lub miedziana blacha służy nam do wykonania podkładek stykowych pod wkręty (rys.2). Do nich właśnie — jeszcze przed montażem całości lutujemy przewody elektryczne. Przycisk „fire" wykonujemy ze sprężystej blachy stalowej (rys. 3).

Teraz pozostaje tylko skręcić naszą konstrukcję, (pamiętając o luźnym zamocowaniu okrągłej blaszki) i przylutować wtyczkę (rys. 4). Sposób połączenia manipulator — wtyczka:

A — 1
B — 2
C — 4
D — 3
E — 8
F — 8
G — 6

Roman Poznański

WSZYSTKO DLA WSZYSTKICH

JEŚLI JUŻ WIESZ DLACZEGO IBM PC NIE ZASPOKOI WSZYSTKICH TWOICH POTRZEB

zapamiętaj:

Przedsiębiorstwo Polonijno-Zagraniczne IMPOL 1
02-641 Warszawa, ul. Malawskiego 7 tel. 44-12-07 (08)
tlx 817218

oferuje:
Modułowe Systemy (Mikro-) Komputerowe!!!
które również są kompatybilne z IBM PC/XT

Zastosowanie:
— systemy kontrolno-pomiarowe
— systemy laboratoryjne i uruchomieniowe
— sterowniki przemysłowe
— systemy gromadzenia i przetwarzania danych

DANE TECHNICZNE:

Modułowy System Mikroprocesorowy	Modułowy System komputerowy
procesory: 8080A, 8085, Z80, 8088	8080A, Z80; procesory peryferyjne 8048, Z80

bogate oprogramowanie pracujące pod kontrolą systemów operacyjnych:

CP/M 2.2, MP/M, MS-DOS 3.0	CP/M 2.2, MP/M
kaseta 3U, wymiar pakietu 100/160	kaseta 6U, wymiar pakietu 220×233,4

pamięci RAM/ROM do 1 Mb z możliwością podtrzymania bateryjnego
kontrolery pamięci dyskowych i kasetowych
programator pamięci EPROM 2708-27512
we/wy TTL z izolacją galwaniczną
moduł grafiki kolorowej
przetworniki A/C i C/A
interfejs szeregowy i równoległy
liczniki do współpracy z przetwornikami obrotowo-impulsowymi

interfejs pomiarowy IEC-625	sterownik silników krokowych
zegar czasu rzeczywistego	terminal alfanumeryczny
dyski elastyczne 2×720kb	2 generacji

systemy zbudowane z podzespołów najnowszej technologii
możliwość współpracy przy aplikacji systemu
Szczegółowe informacje:

tel. 48-19-26	tel. 48-16-18

K-184

JOWISZ TC 500 I HELIOS:

- DEKODERY PAL do samodzielnego wmontowania (bez użycia lutownicy)
- Naprawy modułów i bloków. Zwrotnice antenowe. Wysyłka pocztą.

Zakład Teleelektroniki, 38-420 Korczyna 336a

D-101

INFORMAX
oferuje

- bogaty wybor programow uzytkowych, gier (również sprzedaż wysyłkowa)
- projektowanie i programowanie systemow na mikrokomputery AMSTRAD

Bezpłatna wysyłka katalogów i informacji 02-791 Warszawa, ul Meander 21 m 20

D-154

ZX SPECTRUM..

Naprawy Interfejsy KEMPSTON, SINCLAIR 03-982 Warszawa ul Meissnera 14 m 1. g 18 — 20.30

D-178

Programy do ATARI.

Andrzej Sliwinski, Mieszka I 2 m. 29. 09-400 Płock.

D-164

WSZYSTKO DLA WSZYSTKICH

Skup i sprzedaż MIKROKOMPUTERÓW

- urządzenia peryferyjne
- osprzęt i oprogramowanie
- sprzęt magnetowidowy

prowadzi nowouruchomiony dział w sklepie 143 WPHW Katowice. Oddział Bytom.
Sklep prowadzi sprzedaż pozarynkową.
Zapraszamy do skl. 143 w Bytomiu, ul. 1-go Maja 12 w godz. 10.00 — 18.00, telefon: 81-39-33.

K-161

INTERAMS AMSTRAD IBM

oferuje

- komputery
- drukarki
- oprogramowanie
- literatura
- programy na zamówienie
- pośrednictwo

Udzielamy gwarancji na komputery zakupione w naszej firmie
x Gry dla młodzieży. W-wa, tel. 38-52-20, Okopowa 22/11.

D-134

PROGRAMY KOMPUTEROWE POCZTĄ

DLA ATARI, AMSTRADA, COMMODORE i SPECTRUM

wysyła Agencja Mikrokomputerowa Sosnowiec P-157, tel. 699-649.

K-76

INFORMAX

ul. Meander 21 m. 20
02-791 Warszawa

oferuje

- szeroki asortyment programów użytkowych, gier (również sprzedaż wysyłkowa)
- projektowanie i programowanie systemów na mikrokomputery AMSTRAD

Bezpłatna wysyłka katalogów i informacji.

D-139

ATARI, SPECTRUM, AMSTRAD

— programy, polskie instrukcje wysyła „MEGABAJT" — Warszawa Paryska 17/29 — tel. 17-76-16

D-135

ATARI BASIC

wersja polska, 31 stron A4, cena 750 zł. Zamówienia przekazem. Waldemar Żurko, 43-100 Tychy, Nałkowskiej 24/10.

D-138

ZX SPECTRUM — SERVICE —

Agencyjny Zakład SPHW
nr 751
Warszawa, ul. Mokotowska 61
tel. 28-20-27 11⁰⁰ — 19⁰⁰
rachunki, gwarancja.

D-143

eurobit

Zanim kupisz komputer
Zadzwoń
28-01-76

Kupiłeś komputer
Zadzwoń
28-01-76

PORADY:
— WYBÓR SPRZĘTU
— ZASTOSOWANIA
OPROGRAMOWANIE
INSTRUKCJE
POŚREDNICTWO
ZLECENIA

ATARI
COMMODORE C-16, 116, 4
AMSTRAD
VIC 20; C-64
SHARP
MSX
IBM

WYSTAWIAMY RACHUNKI DLA INSTYTUCJI
Al. Ujazdowskie 18 m. 14 Warszawa w g. 9-18.

POCZTĄ!

I NA MIEJSCU
W-WA, PLAC ZBAWICIELA
„MALUCH"
WSZYSTKIE PROGRAMY NA:
ZX SPECTRUM TIMEX
w cenie 100-300 zł.
INF. LISTOWNA: W. P. K. „SPECTRUM",
00-560 W-WA,
MOKOTOWSKA 19
LUB TEL. 25-95-40

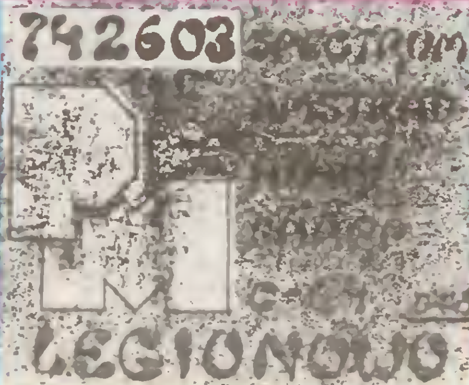
D-128

JAK REKLAMOWAĆ SIĘ W BAJTKU?

Reklamy przyjmuje Młodzieżowa Agencja Wydawnicza (Redakcja Wydawnictw Poradniczych i Reklamy), 04-028 Warszawa, Al. Stanów Zjednoczonych 53, pokój 313. tel.10-56-82

Cena ogłoszeń: 200 zł za 1 cm plus dodatki za kolor

Bajtek 12/86 J.P.



SERWIS KOMPUTEROWY

PROGRAMY. INSTRUKCJE.
Commodore, Atari, Sinclair
Amstrad, Urządzenia peryferyjne
02-383 Warszawa,
Grójecka 128
tel. 46-39-96

D-117

SPRZĘŻENIE ZWROTNE

INDYWIDUALNY BANK DANYCH

Nazywam się **Marek Kawalec.** Jestem uczniem LO, mam 16 lat. Posiadam komputer TIMEX Sinclair 2068, magnetofon MK 232P oraz Atari Pro-Line Joystick. Oprogramowanie: bardzo liczne programy matematyczne i fizyczne, programy muzyczne (BEEP), kilka gier. Interesuję się programowaniem. Chciałbym wymieniać programy oraz informacje o programowaniu. Adres: ul. Katalońska 1 m. 24, 02-763 Warszawa

Jan Krzyżak, uczeń, 19 lat. Mikrokomputer Aquarius (20k). Około 15 programów (głównie gry). Zainteresowania: matematyka, chemia, filatelistyka i wędkarstwo. Wymiana programów. Adres: Żmiąca 69, 34–603 Ujanowice.

Piotr Dulak, student informatyki na Politechnice Wrocławskiej, 20 lat. Mikrokomputer ENTERPRISE — 64. Oprogramowanie:
— assembler napisany w BASIC'u (z możliwością etykietowania)
— słownik polsko-angielski i angielsko-polski
— słownik angielski czasowników nieregularnych oraz test sprawdzający ich znajomość
— mini syntetyzator z pamięcią melodii, autorytmem
— drobne programy graficzne.
Zainteresowania: fotografika, narciarstwo. Adres: ul. Robotnicza 8-12/7, 45–352 Opole

Marcin Szybiak, uczeń, 12 lat. Mikrokomputer Cobra 1 — opracowany przez AV. Zainteresowania: informatyka, budowa i działanie komputerów. Adres: L. Hertz 18, 05–400 Otwock.

Dominik Kołcio, uczeń technikum, 17 lat. Mikrokomputer TOSHIBA MSX. Oprogramowanie: grafika, alfabet Morsa, mapa Polski, biorytmy. Chciałbym uzyskać jakiekolwiek informacje dotyczące programowania mojego mikrokomputera na kodzie maszynowym. Interesuję się informatyką, elektroniką, literaturą fantastyczną i przygodową. Adres: ul. Baciarellego 16/12, 51–649 Wrocław.

Cezary Wacławek, fotograf, 28 lat. Mikrokomputery: ACORN-ELECTRON i LAMBDA 83, sterownik μP SA–80 (na procesorze Z–80), programator EPROM (konstrukcja własna), interface „Plus 1" do ACORN-ELECTRON. Oprogramowanie: do ACORNA — „Klijentlist" do obsługi zakładu FOTO, parę prostych graficznych programów oraz „Rysplyt" czyli rysowanie dwustronnej płytki drukowanej; do LAMBDY 83 — „Kółko i krzyżyk", „Master mind" dowolnego formatu. Zainteresowania: elektronika, informatyka, malarstwo, fotografia. Proponuję założenie „Klubu Użytkowników Mało Popularnych Komputerów". Adres: ul. Lotników 11/110, 05–100 Nowy Dwór Mazowiecki.

Tadeusz Zaremba, inż. budownictwa wodnego, 35 lat. Mikrokomputer SHARP MZ-731, wbudowany magnetofon i drukarka. Proste gry, programy zręcznościowe, obliczeniowe programy projektowe i kosztorysowe, nauka programowania w BASIC'u dla SHARP serii MZ. Zainteresowania: sport, fotografia, muzyka, technika mikrokomputerowa i jej zastosowanie. Adres: ul. Kasztelańska 30/2, 58–314 Wałbrzych.

Robert Nejman, student, 23 lata. Mikrokomputer SINCLAIR QL, (drukarka Brother HR5, mysz. Pakiet programów numerycznych, program kopiujący z ekranu na drukarkę, znaczna ilość drobnych programów ułatwiających współpracę z komputerem. Zainteresowania: zastosowania obliczeniowe komputerów, inżynieria chemiczna, alpinizm, brydż. Adres: ul. Międzynarodowa 60 m. 15, 03-922 Warszawa.

Maciej Izdebski, inż. ceramik, 36 lat. Mikrokomputer DRAGON 32, monitor, programy firmowe. Proponuję wymianę doświadczeń i literatury. Adres: ul. Łużycka 49 m. 1 Kraków.

Drogi Bajtku!

Czy ZX-81 kosztuje tak mało dlatego, że jest to jeden z bardzo prostych komputerów, czy dlatego, że nie ma do niego oprogramowania?
Czy programy dla ZX Spectrum mogą być używane na ZX-81?

(nazwisko i adres do wiadomości redakcji)

Chciałbym kupić komputer Sinclair ZX-81. Czy warto go kupić, czy oszczędzać na inny? Jakie są zalety i wady tego komputera?

Michał Duda
ul. Kardynała Stefana Wyszyńskiego 6/91
10-457 Olsztyn

ZX-81 dawno już przestał liczyć się na zachodnim rynku. Jego niska cena na wolnym rynku w Polsce świadczy o tym, że i u nas coraz szybciej traci on będzie popularność. Sądzę zresztą, że słusznie, jest to urządzenie nie spełniające, moim zdaniem, podstawowych wymagań użytkowych. Zbyt mała i niewygodna klawiatura, brak grafiki wysokiej rozdzielczości tylko 1 KB pamięci o swobodnym dostępie (o ile nie wyposażymy ZX-81 w pakiet rozszerzający pamięć) — to w chwili obecnej bardzo poważne niedostatki. Nawet rezydujący BASIC tego komputera pozbawiony jest pewnej ilości podstawowych funkcji. Zaletą pozostaje w zasadzie tylko wspomniana niska cena.
Również ilość dostępnego w Polsce oprogramowania na ten komputer będzie spadała: import nowych programów z zagranicy w zasadzie już zakończył się, zaś doświadczonych użytkowników ZX-81, mogących zaoferować większe ilości programów, będzie w kraju coraz mniej.
Programy dla Spectrum nie mogą być używane na ZX-81 bez włożenia sporej pracy w ich przystosowanie, nawet w przypadku stosunkowo prostych programów w BASIC-u.
Reasumując, proponowałbym decydując się na zakup pierwszego komputera, rozważyć poważnie możliwość wyboru innego sprzętu.

Czy da się rozszerzyć pamięć ZX Spectrum 48 do 64, a jeśli to możliwe, gdzie kupić taką przystawkę?
Jak uzyskać nieśmiertelność w grach Knight Lore i Alien 8?

Sławomir Bujdo
Os. 1000-lecia 49/20
44-224 Knurów

Spectrum w istocie jest wyposażony w 64 KB pamięci = 16 KB ROM+48 KB pamięci o swobodnym dostępie. Ewentualne dalsze rozszerzenie pamięci RAM może polegać na zainstalowaniu przełączanych banków pamięci. Jedną z możliwości opisuje „Komputer" nr 6/1986.
Sposoby uzyskania nieskończonego życia w wymienionych grach przypomniam za Bajtkiem nr. 8/86:
Knight Lore — POKE 53567,0
Alien 8 — POKE 51736,0

Proszę o informację jaka jest dostępność oprogramowania do C116 w naszym kraju i w ogóle na Zachodzie. Czytałem, że ten MK był „niewypałem" rynkowym, więc może lepszym wyjściem byłaby jego sprzedaż i kupno innego urządzenia, np. ZX Spectrum. Interesuje mnie, czy istnieją wersje najbardziej popularnych gier komputerowych na C116.

Wasz Czytelnik

Komputer Commodore C116 jest wersją komputera Commodore 16. Podstawową zmianą jest tu użycie tańszej w produkcji (i znacznie gorszej) klawiatury oraz zmienionej obudowy. Modyfikacje te wpłynęły na obniżenie ceny — z innymi modelami mikrokomputerów C116 konkurował i konkuruje nadal przede wszystkim wyjątkowo niską ceną.
Oba modele reklamowane były jako „komputer do nauki języka BASIC", podobnie jak kilka lat wcześniej model VIC-20. Rzeczywiście, pamięć stała tych maszyn zawiera interpreter rozszerzonej znacznie wersji języka. Siłą rzeczy jednak, wobec wyposażenia tylko w 16 K pamięci RAM, oraz nasycenia rynku komputerem bardziej uniwersalnym, jakim niewątpliwie był Commodore 64, oba modele nie osiągnęły wielkiej popularności.
Mała pojemność pamięci jest również powodem, dla którego nie znajdziemy dla C116 gier o tym stopniu złożoności, jak dla Spectrum czy C64.
W porównaniu z ZX Spectrum 16 K omawiany komputer stanowi, jak sądzę, lepszy standard. Rozbudowa pamięci do 64 K jest wszakże problematyczna ze względu na szczupłość miejsca wewnątrz obudowy.
Oprogramowanie na C116 nie występuje w obfitości. Wydaje się, że w warunkach krajowych możliwe jest skompletowanie kilkudziesięciu programów.
Komputer ten jest przystosowany do współpracy z magnetofonem takim samym, jak do innych maszyn Commodore. Kwestię dostosowania własnego magnetofonu omówiono w „Bajtku" nr 3/85.

W numerze 3-4 „Bajtka", na stronie 22, w programie Piotra Sypniewskiego wkradł się chyba błąd. W wierszu nr 40. Przy wprowadzaniu linii po liczbie 0.26 pojawia się znak zapytania. Czy mogę prosić o sprawdzenie i informację, czego błąd dotyczy?

Krzysztof Więczorek
ul. Grunwaldzka 116/19
80-244 Gdańsk

Rzeczywiście. Pomiędzy liczbami 0 i 26 powinien występować przecinek, a nie kropka. Interpreter BASIC-a, sprawdzając poprawność składniową wiersza rozpoznał ciąg „0.26" jako jedną liczbę, co wobec faktu, że parametrem PRINT AT musi być zawsze parą liczb, spowodowało zasygnalizowanie błędu.
Istotne funkcje tego krótkiego programiku zostaną zachowane również przy pominięciu linii 40. W przypadku jej przepisania, powinna jednak wyglądać tak:

40 PRINT AT 0,26; "Rys."; "-";n

Posiadam symulator lotu na Commodore 64. Chciałbym dowiedzieć się o programach tego typu (dostępnych na rynku), a w szczególności o typach samolotów, które to programy pozwalają „pilotować". Czy zawsze jest to typ „nieokreślony", czy też programy te są odpowiednikami prawdziwych samolotów?

Piotr Drynski
Dęblin

Dla Commodore 64 napisano do tej pory wiele różnych symulatorów lotu. Zazwyczaj ich producenci powołują się na istniejący pierwowzór. Oto kilka przykładów
Flight Simulator II — symulacja lotu samolotem Piper Cherokee Archer
Flight 747 — symulacja lotu samolotem Boeing 747
Flight 727 — to samo dla Boeinga 727
Jet Combat Simulator — lot myśliwcem F-15 Eagle
Jet — mamy dwie możliwości wyboru samolotu
General Dynamics F-16 lub
F-18 w wersji morskiej (start i lądowanie na lotniskowcu)
Stunt Flyer — akrobacyjny Piper Cub
Super Huey — helikopter Bell Huey Cobra
Dam Buster — samolot bombowy Avro Lancaster; ten program odtwarza atmosferę słynnej operacji bombardowania zapór wodnych w Zagłębiu Ruhry w 1942 roku

Marcin

KLAN NIETYPOWYCH

Podczas tegorocznych wakacji we Francji kupiłem mikrokomputer „SEGA 3000". Chciałbym korzystać z pomocy tego sprzętu ale niestety muszę się przyznać iż w tych sprawach jestem „zielony". Może ktoś z czytelników posiada nieco większą wiedzę i praktykę w obsługiwaniu tego typu mikrokomputera i udzieli mi kilku porad.

Zbigniew Kopicz
Tylice
ul. Świerczewskiego 13

Od 2 lat posiadam komputer Oric-Atmos. Początkowo miałem tylko kasetę demonstracyjną. Obecnie posiadam już ok. 150 programów. Są to gry, programy użytkowe, rozszerzenie Basica, muzyczne itp.

Zbigniew Siatkowski
ul. Bazyliańska 8 m.146
03-203 Warszawa



Na listy czytelników odpowiada Marcin Waligórski

KIERUNEK: KSIĘŻYC!

Dokonczenie ze str. 32

wadzenie przerasta możliwość automatów czy uczestników krótkoterminowych ekspedycji na powierzchnię Księżyca. Ale badacze, przez długi okres czasu żyjący w stałej księżycowej bazie naukowej, odpowiednio „uzbrojeni" w aparaturę naukową, mogliby zajmować się szeroką gamą problemów.

Specjaliści widzą kilka głównych kierunków wykorzystania Księżyca i księżycowych materiałów przy zagospodarowywaniu wysokich orbit okołoziemskich. Za jeden z najważniejszych kierunków doktor nauk matematyczno-fizycznych Władysław Szewczenko (z którego materiałów tu korzystam) uważa pełniejsze wykorzystanie energii słonecznej. Opracowano projekt stworzenia na orbicie geostacjonarnej urządzeń energetycznych zdolnych tanio wytwarzac energię elektryczną i przesyłać ją na Ziemię w formie przetworzonej (np. w postaci mikrofal) lub odbijać na Ziemię światło słoneczne. Na tej podstawie proponuje się rozwinąć opartą na nowych zasadach energetykę, a także nowe kosmiczne technologie i zasadniczo nowe metody w agrotechnice. Precyzyjne obliczenia wykazały jednak, że bez wykorzystania księżycowych zasobow naturalnych i księżycowego przemysłu urzeczywistnienie podobnych projektów jest praktycznie niemożliwe. W czym rzecz?

Otóż nakłady energetyczne niezbędne dla przetransportowania danego ładunku z Księżyca na wysoką orbitę wokółziemską są... 20-30 razy mniejsze, niż przy sprowadzeniu tej samej masy ładunku z Ziemi. Na przykład, masa użyteczna przy starcie systemu kosmicznego wielokrotnego wykorzystania z kosmodromu ziemskiego stanowi dotychczas zaledwie około 1,5 procent całej startującej masy. Przy starcie z Księżyca ładunek użyteczny może natomiast stanowić nawet 50 proc. ogolnej masy startującego systemu.

TANIEJ NIŻ „APOLLO"

W przyszłości, gdy wokoł Ziemi zaczną powstawac wielkie stacje kosmiczne z kilkudziesięcioosobowymi załogami, będą się tam musiały również znaleźc setki ton materiałow niezbędnych dla stworzenia pewnej niezawodnej ochrony tych zamieszkałych obiektów przed skutkami promieniowania kosmicznego. Bloki, utworzone z gruntu księżycowego, z powodzeniem mogą spełnic funkcję takiego ochronnego pokrycia. Obliczono, że dla przygotowania osłony o wiele racjonalniej jest wykorzystywac grunt dostarczany z Księżyca, niż wystrzeliwac z Ziemi moduły z gotowymi fragmentami osłony czy też sprowadzać z Ziemi materiały niezbędne dla przygotowania tej osłony już na orbicie.

Oczywiscie, doprowadzenie do trwałego zamieszkania ludzi na innym ciele kosmicznym wymaga olbrzymich nakładów. Jednak stałe doskonalenie techniki kosmicznej, potanienie jej wytwarzania doprowadzi do tego, że w momencie gdy możliwe się już stanie rozpoczęcie realizacji tego projektu, faktyczne nakłady nie przekroczą tych, jakie zostały w swoim czasie wyłożone na pilotowane ekspedycje księżycowe z serii „Apollo". Przy współpracy międzynarodowej nakłady każdego kraju będą oczywiscie odpowiednio mniejsze. A — zapytajmy na koniec — czy można ocenić ten pożytek dla ludzkości, jaki odniosą wszystkie panstwa z pokojowego zagospodarowania kosmosu? „Gwiezdny pokój" będzie służył wszystkim, gdy tymczasem — jako alternatywa — „gwiezdne wojny" mają prawie wszystkim, z wyjątkiem Amerykanów, szkodzic!

Waldemar Siwiński

DWA PALCE KOMPUTERA

Z dziesiętnym systemem zapisywania liczb zżyliśmy się tak, że na ogół nie zastanawiamy się nad zasadą jego funkcjonowania, pozwolę więc sobie przypomnieć kilka podstawowych faktów. W systemie tym mamy dziesięc znaków (cyfr): 0, 1, 2,...,9, używanych do zapisu liczb. Aby przy pomocy tylko dziesięciu cyfr zapisywać dowolnie duże liczby używamy systemu pozycyjnego — cyfry na kolejnych pozycjach oznaczają ile razy w liczbie występuje odpowiednia potęga liczby 10.

Dla przykładu popatrzmy na liczbę 1804:

1	8	0	4
10^3	10^2	10^1	10^0

Ciąg cyfr 1804 jest skroconym zapisem faktu, że chodzi nam o liczbę następującą. $1 * 10^3 + 8 * 10^2 + 0 * 10^1 + 4 * 10^0$.

Liczba 10 jest podstawą systemu, jej kolejne potęgi wyznaczają wagi poszczególnych pozycji. Zapamiętajmy jeszcze, że mamy dokładnie tyle samo (10) cyfr, ile wynosi podstawa systemu.

Podstawą systemu liczenia wcale nie musi byc 10, jako przykład wybierzmy liczbę 8. Mamy wtedy osiem cyfr (od 0 do 7), a kolejne pozycje wyznaczone są przez potęgi osemki: 8^0, 8^1, 8^2, itd. Np. osemkowy zapis 132 oznacza $1 * 8^2 + 3 * 8^1 + 2 * 8^0$, czyli dziesiętnie 90 (bo $1 * 64 + 3 * 8 + 2 * 1 = 90$). Bardzo to nienaturalne, niemniej jednak system ósemkowy (inaczej oktalny) był kiedys bardzo popularny w informatyce.

Skoro podstawą systemu liczenia mogą być rożne liczby, to czemu ludzie wybrali akurat 10? Byc może stało się tak ponieważ pierwszym przyrządem do liczenia były palce, a mamy ich własnie dziesięc.

Jesli jest to prawdą, to niewykluczone, że komputery mają tylko po dwa palce, gdyż zapisują liczby w systemie dwójkowym (systemie o podstawie 2). Tym z Was, ktorzy mają problemy z systemem dwojkowym, inaczej zwanym binarnym, radzę przyjrzec się uważnie systemowi dziesiętnemu, gdyz zasada w obu jest taka sama

W systemie dwójkowym mamy dwie cyfry 0 i 1. Kolejne pozycje w liczbie odpowiadają kolejnym potęgom dwojki — patrząc z lewa na prawo mamy: 2^0, 2^1, 2^2, 2^3, itd

I znów przykład.

Liczba binarna:	1	0	1	1
Znaczenie pozycji:	2^3	2^2	2^1	2^0

Ciąg cyfr 1011 w tym przypadku jest zapisem faktu, że chodzi nam o liczbę będącą sumą:

$1 * 2^3 + 0 * 2^2 + 1 * 2^1 + 1 * 2^0$

W tym przypadku z gory wiedzielismy, że mamy do czynienia z liczbą dwojkową. Jesli jednak liczby binarne i dziesiętne występują obok siebie, to te pierwsze zaznaczamy pisząc na koncu 2, lub B, np.: 1011_2 lub 1011B.

Aby upewnic się, że wszystko jasne popatrzmy raz jeszcze na oba systemy:

Dziesiętne: 4 3 0 7 czyli
$4 * 10^3 + 3 * 10^2 + 0 * 10^1 + 7 * 10^0$
1 1 0 1 czyli
$1 * 10^3 + 1 * 10^2 + 0 * 10^1 + 1 * 10^0$
10^3 10^2 10^1 10^0

Binarnie 1 1 0 1 czyli
$1 * 2^3 + 1 * 2^2 + 0 * 2^1 + 1 * 2^0$
2^3 2^2 2^1 2^0

Dla liczb dwójkowych obowiązują zasady wykonywania działan arytmetycznych podobne do zasad dla liczb dziesiętnych, ale o to niech się martwią komputery i ich konstruktorzy. Dla nas ważny jest fakt, że każdą liczbę dziesiętną można zapisać w postaci dwójkowej. Co więcej jest na to bardzo prosty sposob. Dzielimy liczbę dziesiętną przez 2. Albo dzieli się dokładnie (reszta z dzielenia 0), albo zostanie reszta 1. Otrzymaną resztę 0 lub 1, zapisujemy jako pierwszą cyfrę liczby binarnej. Wynik dzielenia znów dzielimy przez 2, resztę dopisujemy z lewej strony poprzedniej cyfry itd. dopoki jest co dzielic.

Przykład: znaleźc binarną reprezentację liczby 57.

57 : 2 = 28	reszta 1	1
28 : 2 = 14	reszta 0	01
14 : 2 = 7	reszta 0	001
7 : 2 = 3	reszta 1	1001
3 : 2 = 1	reszta 1	11001
1 : 2 = 0	reszta 1	111001

Reprezentacją binarną liczby 57 jest 111001.

Sprawdźmy, czy wynik jest poprawny, przechodząc z powrotem na reprezentację dziesiętną. Metoda jest rownież bardzo prosta. Trzeba zapisac liczbę dwójkową w postaci sumy odpowiednich potęg dwojki (wygodniej jest to robić w kolejności odwrotnej niż w zapisie liczby).

$111001 = 1 * 2^0 + 0 * 2^1 + 0 * 2^2 + 1 * 2^3 + 1 * 2^4 + 1 * 2^5 = 1 + 0 + 0 + 8 + 16 + 32 = 57$

Jak widac nasza konwersja działa poprawnie. Żeby przebiegała rownież sprawnie warto znać wartosci kolejnych potęg dwojki. Łatwo je obliczyć, pamiętając, że:

$2^0 = 1$
$2^1 = 2 * 2^0 = 2$
$2^2 = 2 + 2^4 = 4$
$2^3 = 2 * 2^2 = 8$
$2^4 = 2 * 2^3 = 16$ itd

Andrzej Pilaszek



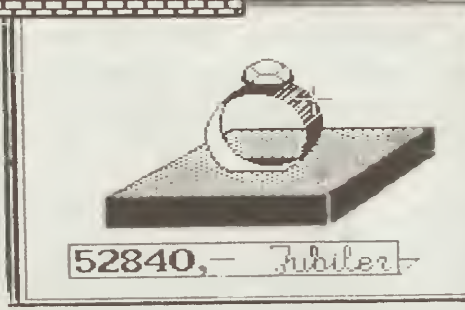
* Konkurs świąteczny

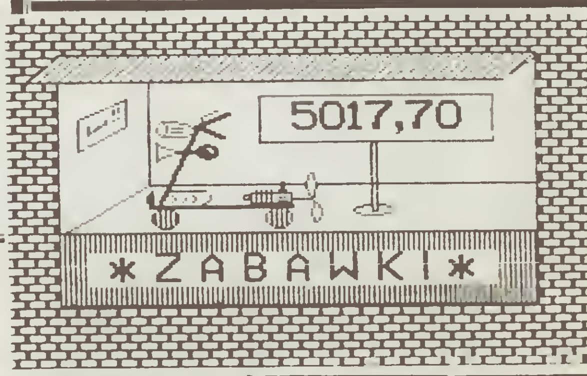
Autorzy: R. Poznański, M. Silski
Commodore 64, Newsroom,
Gemini 10 X, VC 1541

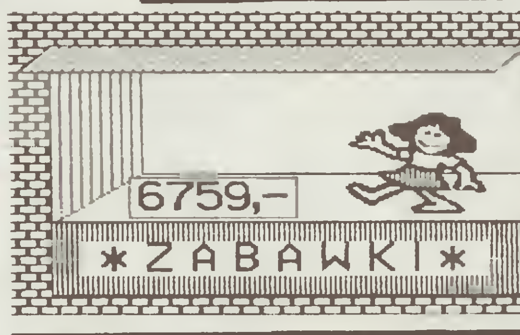
MAW
Bajtek – tylko 100 zł!

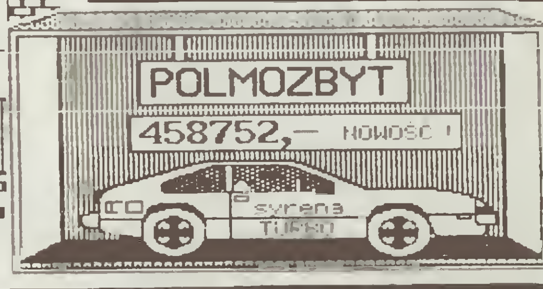












ŚWIĄTECZNY KONKURS „BAJTKA"

Nasz przyjaciel Kubus wygrał w Totolotka milion (powiedzmy prawie milion) i postanowił zrobić prezenty swoim najbliższym. Wszystkie pieniądze powkładał do kopert w ten sposób, by mógł zapłacić dokładnie każdą sumę pełnymi kopertami. Kubus postanowił kupić w pierwszej kolejności prezent dla mamy, następnie dla taty, później dla narzeczonej, siostry i brata. Jednak wychodząc z domu zobaczył w sklepie firmowym Młodzieżowej Agencji Wydawniczej coś [illegible] — choć to niezbyt grzeczne — rozpoczął zakupy od prezentu dla siebie.

Popatrzcie na rysunek. Widzicie sklepy, które odwiedził Kubus, widzicie także koperty z pieniędzmi. Spróbujcie odgadnąć, jaka była kolejność zakupów i którymi kopertami płacił Kubus za poszczególne przedmioty. Zwróćcie uwagę, że sumy na poszczególnych kopertach to kolejne potęgi dwójki pomnożone przez 10 groszy. Jeśli więc zapiszecie ceny prezentów jako dwójkową liczbę dziesięciogroszówek, to reszta będzie już prosta. (O systemie dwójkowym przeczytać możecie w tym numerze). Ustawcie koperty w kolejności zakupów. Koperty, którymi Kubuś płacił za ten sam przedmiot, ustawcie od największej do najmniejszej. Teraz przepiszcie litery umieszczone na kopertach do diagramu i odczytajcie hasło.

Na tych, którzy do 31 stycznia 1987 roku nadeślą prawidłowe rozwiązanie oraz dołączą kupon konkursowy, czeka mikrokomputer Atari 800 XL ufundowany przez Przedsiębiorstwo Zagraniczne „KAREN" oraz w[illegible] n[illegible]ch [illegible] nagród.

Bajtek

	0.1 Ę	0.2 A	0.4 I
0.8 T	1.6 A	3.2 U	6.4 P
12.8 M	25.6 O	51.2 K	102.4 K
204.8 W	409.6 N	819.2 A	1638.4 D
3276.8 Z	6553.6 G	13107.2 R	26214.4 A
52428.8 I	104857.6 R	209715.2 E	419430.4 T

CHOINKA

Cześć Maluchy!

Święta za pasem, najwyższy czas zabrać się za ubieranie choinki. Zanim jednak stanie w waszym domu prawdziwa choinka, warto trochę potrenować. Oczywiście pomoże wam komputer.



Na naszej komputerowej choince zawieszać będziemy tylko bombki i gwiazdki. Nie będzie to jednak takie łatwe. Otóż bombka jest dwa razy cięższa niż gwiazdka i — aby nasza choinka nie przewróciła się — musimy powiesić ją na gałązce dwa razy dalej. Dlaczego właśnie tak? Popatrzmy na rysunek.

rys. 1

Dwóch ludzików stoi na huśtawce. Obydwaj ważą tyle samo i znajdują się w tej samej odległości od środka huśtawki. Dzięki temu nie przechyla się ona w żadną stronę. Ale oto ludzik z prawej strony zrobił kilka kroków do przodu



rys. 2

i chociaż jego waga pozostała bez zmiany, huśtawka przechyliła się w lewo. Teraz z kolei na huśtawkę wszedł trzeci ludzik.



rys. 3

i znów powróciła równowaga.

Jeśli nie jesteś pewny czy tak się stanie, możesz wybrać się z dwoma kolegami na najbliższy plac zabaw i przeprowadzić to doświadczenie.

W fizyce nazywa się to zasadą dźwigni dwustronnej, lub — jeszcze mądrzej — równowagą momentów sił. Prawa te mówią, że każdy ciężar możemy zrównoważyć na huśtawce ciężarem o połowę mniejszym, jeśli umieścimy go dwa razy dalej od środka. Na tym właśnie będzie polegała nasza zabawa. Gwiazdka musi być zawieszona na gałązce o połowę bliżej pnia niż bombka. Jeśli się pomylimy, choinka przewróci się.

Bądźcie więc ostrożni!

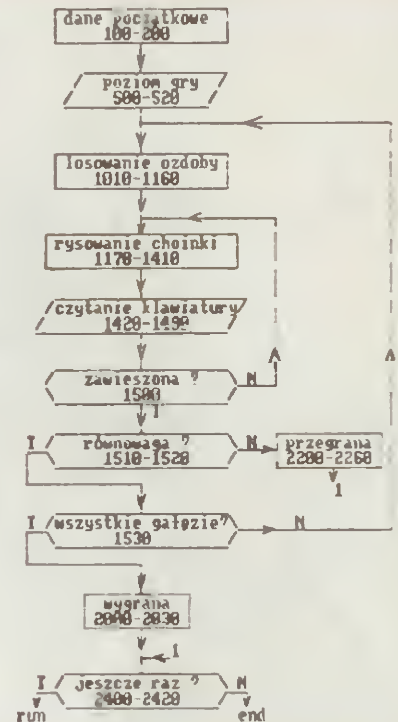
Romek

```
1 REM ****** COMMODORE 64, VC 20 ******
100 PRINT " CLR ": REM  CLR  oznacza
520 PRINT " CLR ": REM  nacisniecie
521                REM  klucza CLR/HOME
1170  PRINT" HOME ":REM  HOME  oznacza
1420  GET K$       :REM nacisniecie
2200 PRINT" HOME " :REM kluczy shift +
2201                REM CLR HOME
```

```
1 REM *********** Spectrum ***********
110 DIM l$(7,1)
120 DIM p$(7,1)
1020  LET los=RND
1080  LET pozprawa(b)=INT(RND*(b+2))+1
1150  LET pozprawa(b)=INT(RND*(b+2))+1
1170 PRINT AT 0,0;
2200 PRINT AT 0,0;

Zmiana nazw tablic:
ozdobaprawa$ --- p$
ozdobalewa$  --- l$
```

```
1 REM ************ MERITUM ************
10 RANDOM
1020  LET los=RND(0)
1080  LET pozprawa(b)=INT(RND(0)* b+2))
+1
1150  LET pozprawa(b)=INT(RND(0)* b+2))
+1
1170 PRINT @1,;
2200 PRINT @1,;
```




```
1 REM ******* Amstrad-Schneider *******
99 REM * deklaracje tablic i zmiennych *
100 CLS
110 DIM ozdobalewa$(7)
120 DIM ozdobaprawa$(7)
130 DIM pozlewa(7)
140 DIM pozprawa(7)
150   FOR i=1 TO 7
160   LET ozdobalewa$(i)=" "
170   LET ozdobaprawa$(i)=" "
180   LET pozlewa(i)=10
190   LET pozprawa(i)=1
200   NEXT i
499 REM ****** stopien trudnosci *******
500 INPUT "Jakie chcesz tempo gry (1-9)"
;tempo
510 LET limit=3000/tempo
520 CLS
999 REM ************* gra **************
1000   FOR b=1 TO 5
1009   REM ** parametry danej galezi **
1010   LET czas=0
1020   LET los=RND(1)
1030   IF los>0.5 THEN GOTO 1110
1040   LET ozdobalewa$(b)="o"
1050   LET wagalewa=1
1060   LET ozdobaprawa$(b)="*"
1070   LET wagaprawa=2
1080   LET pozprawa(b)=INT(RND(1)*(b+2))
+1
1090   IF pozprawa(b)>(b+2)/2 THEN GOTO
1080
1100   GOTO 1170
1110   LET ozdobalewa$(b)="*"
1120   LET wagalewa=2
1130   LET ozdobaprawa$(b)="o"
1140   LET wagaprawa=1
1150   LET pozprawa(b)=INT(RND(1)*(b+2))
+1
1160   IF pozprawa(b)/2<>INT(pozprawa(b)
/2) THEN GOTO 1150
1169   REM ***** rysowanie choinki *****
1170   LOCATE 1,1
1180   PRINT "             A"
1190   PRINT "             X           *=
o+o"
1200     FOR d=3 TO 7
1210     LET p=1:LET k=12-d: LET z$=" "
1220     GOSUB 2600
1230     LET k=d: LET z$="<"
1240     GOSUB 2600
1250     PRINT "X";
1260     LET z$=">"
1270     GOSUB 2600
1280     PRINT
1290     LET k=12-pozlewa(d-2): LET z$="
 "
1300     GOSUB 2600
1310     PRINT ozdobalewa$(d-2);
1320     LET k=pozlewa(d-2)-1
1330     GOSUB 2600
1340     PRINT "X";
1350     LET k=pozprawa(d-2)-1: LET z$="
 "
1360     GOSUB 2600
1370     PRINT ozdobaprawa$(d-2);
1380     PRINT
1390     PRINT "             X"
1400     NEXT d
1410   PRINT "           --/--"
1419   REM *** zmiana pozycji ozdoby ***
1420   LET k$=INKEY$
1430   LET czas=czas+1
1440   IF czas>limit THEN GOTO 2200
1450   IF k$="" THEN GOTO 1420
1460   IF k$="z" THEN LET pozlewa(b)=poz
lewa(b)+1
1470   IF k$="x" THEN LET pozlewa(b)=poz
lewa(b)-1
1480   IF pozlewa(b)<1 THEN LET pozlewa(
b)=1
1490   IF pozlewa(b)>12 THEN LET pozlewa
(b)=12
1500   IF k$<>" " THEN GOTO 1170
1509   REM *** obliczenie rownowagi ***
1510   IF wagalewa*pozlewa(b)=wagaprawa*
pozprawa(b) THEN GOTO 1530
1520   GOTO 2200
1530   NEXT b
1997 REM
1998 REM ********* podprogramy *********
1999 REM *********** wygrana ***********
2000   FOR i=1 TO tempo
2010   PRINT "# ";
2020   NEXT i
2030 PRINT "To prezenty dla ciebie!"
2040 GOTO 2400
2199 REM ********** przegrana **********
2200 LOCATE 1,1
2210   FOR i=1 TO 16
2220   PRINT "                        "
2230   NEXT i
2240 PRINT "        %=)(&&'""
2250 PRINT "     <<><+$##!#()"
2260 PRINT"Koniec zabawy!!!"
2399 REM ******* jeszcze raz? *******
2400 INPUT "Grasz jeszcze raz";odp$
2410 IF odp$="t" THEN RUN
2420 END
2599 REM ******* fragment galezi *******
2600   FOR i=p TO k
2610   PRINT z$;
2620   NEXT i
2630 RETURN
```

KIERUNEK: KSIĘŻYC!



„Gwiezdny pokój" zamiast „gwiezdnych wojen" — tak można najkrócej określic propozycję Związku Radzieckiego wspólnego podjęcia przez wszystkie państwa wieloetapowego programu zagospodarowania kosmosu w perspektywie do 2000 roku. Proponuje się między innymi, aby w okresie tym stworzyć warunki umożliwiające już w pierwszych latach XXI wieku praktyczne zagospodarowanie i wykorzystanie Księżyca, jako bazy dla wysyłania ekspedycji do innych planet.

TAJEMNICA „NIERDZEWNEGO" ŻELAZA

Przydatnosć naukowa stworzenia długofunkcjonującej zamieszkałej stacji badawczej na powierzchni Księżyca nie budziła nigdy zastrzeżeń. Występują tam bowiem unikalne warunki naturalne stanowiące klucz do zrozumienia wczesnej historii Ziemi.

Z Księżyca, na jakościowo nowych podstawach, można badać dalszy i bliższy kosmos. Wysoka próżnia i mała siła ciężkosci pozwalają przeprowadzac unikalne eksperymenty z zakresu fizyki, chemii, biologii i innych nauk. Dalszych badan wymagają niektore problemy wynikłe przy poznawaniu na Ziemi własciwosci gruntu księżycowego dostarczonego przez aparaty kosmiczne. Na przykład, na czym polega tajemnica „nierdzewnego" księżycowego żelaza? Dlaczego posadzone w księżycowy grunt rosliny rozwijają się w porównywalnych warunkach o wiele intensywniej niż na zwykłej ziemskiej glebie? Co jest przyczyną namagnesowania księżycowych skał? Takich pytań jest o wiele więcej.

Na Księzycu nie ma praktycznie atmosfery, dlatego zachowały się tam w stanie niezmienionym ślady niewiarygodnie dalekich wydarzeń, mających miejsce, możliwe, w pierwszych 500 milionach lat istnienia Systemu Słonecznego. Oddzielne fragmenty dostarczonych na Ziemię skał księżycowych mają, według niektorych ocen, taki własnie wiek. Cząsteczki wchodzące w skład strumienia wysyłanej przez Słońce plazmy pozostawiają mikroskopijne ślady-tory w ziarenkach piasku wierzchniej warstwy rozdrobnionych księżycowych skał. Własnie na podstawie tych śladów zostawionych w najstarszych skałach, można odtworzyć historię słonecznej aktywności w ciągu ostatnich 3-4 miliardów lat. Badania te mają niezwykłą wagę dla poznania mechanizmu pracy naszej „latarni".

METEORYTY Z ZIEMI

Przy analizie gruntu księżycowego odkryto fragmenty najstarszych skał ziemskich. Na Ziemi, w stanie czystym, takie skały już nie występują. Na Księżycu mogły one przetrwac w stanie niezmienionym miliardy lat. Skąd jednak się tam wzięły?

Na trop odpowiedzi naprowadza nas fakt, że stosunkowo niedawno wykryto na Ziemi fragmenty skał księżycowych, ktore dotarły do nas jako meteoryty. Dlatego, jako w pełni prawdopodobną, można rownież wyobrazić sobie i odwrotną drogę wędrówki materii: z Ziemi na Księżyc. Gdy nasza planeta była jeszcze młoda i nie posiadała tak gęstej atmosfery na jej powierzchnię padały meteoryty o wiele częściej niż obecnie. Każdemu takiemu upadkowi towarzyszył silny wybuch. Niektore z wyrzuconych przez ten wybuch ziemskich skał mogły uzyskac drugą prędkosc kosmiczną i po odpowiednio długiej podroży w kosmosie trafic na Księżyc.

BEZ KSIĘŻYCA ANI RUSZ!

Oczywiście, organizacja badań naukowych jest na Księżycu bardzo złożona. Ich przepro-

Dokończenie na str. 29